150 ptas.

45

MI COMPUTER

CURSO PRACTICO DEL ORDENADOR PERSONAL, EL MICRO Y EL MINIORDENADOR



Editorial Delta, S.A.

MI COMPUTER CURSO PRACTICO
DEL ORDENADOR PERSONAL, EL MICRO Y EL MINIORDENADOR

Publicado por Editorial Delta, S.A., Barcelona

Volumen IV - Fascículo 45

Director: José Mas Godayol
Director editorial: Gerardo Romero
Jefe de redacción: Pablo Parra
Coordinación editorial: Jaime Mardones, Francisco Martín
Asesor técnico: Ramón Cervelló

Redactores y colaboradores: G. Jefferson, R. Ford, F. Martín, S. Tarditti, A. Cuevas, F. Blasco
Para la edición inglesa: R. Pawson (editor), D. Tebbutt (consultant editor), C. Cooper (executive editor), D. Whelan (art editor), Bunch Partworks Ltd. (proyecto y realización)

Realización gráfica: Luis F. Balaguer

Redacción y administración:
Paseo de Gracia, 88, 5.°, 08008 Barcelona
Tels. (93) 215 10 32 / (93) 215 10 50 - Télex 97848 EDLTE

MI COMPUTER, *Curso práctico del ordenador personal, el micro y el miniordenador*, se publica en forma de 96 fascículos de aparición semanal, encuadernables en ocho volúmenes. Cada fascículo consta de 20 páginas interiores y sus correspondientes cubiertas. Con el fascículo que completa cada uno de los volúmenes, se ponen a la venta las tapas para su encuadernación.
El editor se reserva el derecho de modificar el precio de venta del fascículo en el transcurso de la obra, si las circunstancias del mercado así lo exigieran.

ISBN: 84-85822-83-8 (fascículo) 84-7598-005-8 (tomo 4)
84-85822-82-X (obra completa)
Depósito Legal: B. 52/1984

Fotocomposición: Tecfa, S.A., Pedro IV, 160, Barcelona-5
Impresión: Cayfosa, Santa Perpètua de Mogoda (Barcelona) 218411
Impreso en España - Printed in Spain - Noviembre 1984

Editorial Delta, S.A., garantiza la publicación de todos los fascículos que componen esta obra.

Distribuye para España: Marco Ibérica, Distribución de Ediciones, S.A., Carretera de Irún, km 13,350. Variante de Fuencarral, 28034 Madrid.
Distribuye para Argentina: Viscontea Distribuidora, S.C.A., La Rioja 1134/56, Buenos Aires.
Distribuye para Colombia: Distribuidoras Unidas, Ltda., Transversal 93, n.° 52-03, Bogotá D.E.
Distribuye para México: Distribuidora Intermex, S.A., Lucio blanco, n.° 435, Col. San Juan Tlihuaca, Azcapotzalco, 02400, México D.F.
Distribuye para Venezuela: Distribuidora Continental, S.A., Edificio Bloque Dearmas, final Avda. San Martín con final Avda. La Paz, Caracas 1010.

Pida a su proveedor habitual que le reserve un ejemplar de MI COMPUTER. Comprando su fascículo todas las semanas y en el mismo quiosco o librería, Vd. conseguirá un servicio más rápido, pues nos permite realizar la distribución a los puntos de venta con la mayor precisión.

Servicio de suscripciones y atrasados (sólo para España)
Las condiciones de suscripción a la obra completa (96 fascículos más las tapas, guardas y transferibles para la confección de los 8 volúmenes) son las siguientes:

a) Un pago único anticipado de 16 690 ptas. o bien 8 pagos trimestrales anticipados y consecutivos de 2 087 ptas. (sin gastos de envío).

b) Los pagos pueden hacerse efectivos mediante ingreso en la cuenta 6.850.277 de la Caja Postal de Ahorros y remitiendo a continuación el resguardo o su fotocopia a Editorial Delta, S.A. (Paseo de Gracia, 88, 5.°, 08008 Barcelona), o también con talón bancario remitido a la misma dirección.

c) Se realizará un envío cada 12 semanas, compuesto de 12 fascículos y las tapas para encuadernarlos.

Los fascículos atrasados pueden adquirirse en el quiosco o librería habitual. También pueden recibirse por correo, con incremento del coste de envío, haciendo llegar su importe a Editorial Delta, S.A., en la forma establecida en el apartado b).

Para cualquier aclaración, telefonear al (93) 215 75 21.

No se efectúan envíos contra reembolso.

Editorial Delta, S.A.



FINANCIAL MICRO TIMES



Poder para el pueblo

A muchas personas les atrae la idea de llevar su propio negocio. Pero, aun cuando se trate de una tienda pequeña, los problemas que entraña una aventura de este tipo pueden ser enormes. Y esto se aplica en toda su dimensión cuando se trata de llevar las finanzas de su empresa. Sin embargo, en este capítulo no vamos a ocuparnos del software de gestión práctico, sino de una selección de paquetes de juegos que simulan algunos de los problemas y los desafíos que plantean los grandes negocios. El más conocido de estos juegos es el *Monopoly*, que es una adaptación del juego de tablero sobre administración y compra-venta de propiedades. Los juegos de gestión son ideales para el ordenador, dado que no hay ni fichas ni papel moneda que se puedan extraviar y que el ordenador efectúa todos los cálculos.

La gama de empresas que se puede llevar es muy diversa: cervecerías, líneas aéreas, fábricas de automóviles y granjas son sólo algunos ejemplos. Obviamente, cada tipo de actividad tiene sus propios problemas.

Independientemente del tipo de empresa que se elija, los programas de simulación de gestión funcionan de forma muy similar. En todos los casos el jugador es el gerente de una empresa determinada y al comienzo del juego dispone de una cantidad inicial de dinero. En el programa *Corn cropper* (Cosechador de trigo), en el que se dirige una granja, esta cantidad es de apenas £ 50 000; pero si se está a cargo de una compañía petrolera, como en el juego *Dallas*, son necesarios cien millones de dólares sólo para arrancar.

Se invierte luego parte de este capital en lo que se denomina "activo fijo" (aviones, fábricas de automóviles, terrenos, etc.). Valiéndose del mismo se puede comenzar entonces a producir sus artículos o servicios (vuelos, coches, trigo) y obtener ingresos.

El grado de realismo varía de un programa a otro. Algunos de ellos, como *Dallas*, están a todas luces pensados como entretenimiento. Otros pueden llegar a ser muy realistas: el programa *Corn cropper* posee datos acerca de las horas de sol y la cantidad de lluvia, las condiciones ideales para cultivar trigo, los salarios de los obreros contratados y los costos de mantenimiento de los tractores, que son sólo algunos de los factores que deben tener en cuenta los granjeros modernos.

Habrá, inevitablemente, algunas personas a las que no les satisfaga la idea de dirigir meramente una empresa, ni siquiera aunque se trate de una corporación petrolífera de $100 millones. Para estas personas la respuesta evidente consiste en ¡intentar gobernar un país! *1984* es un juego que simula la economía de Gran Bretaña, siendo el jugador el primer ministro.

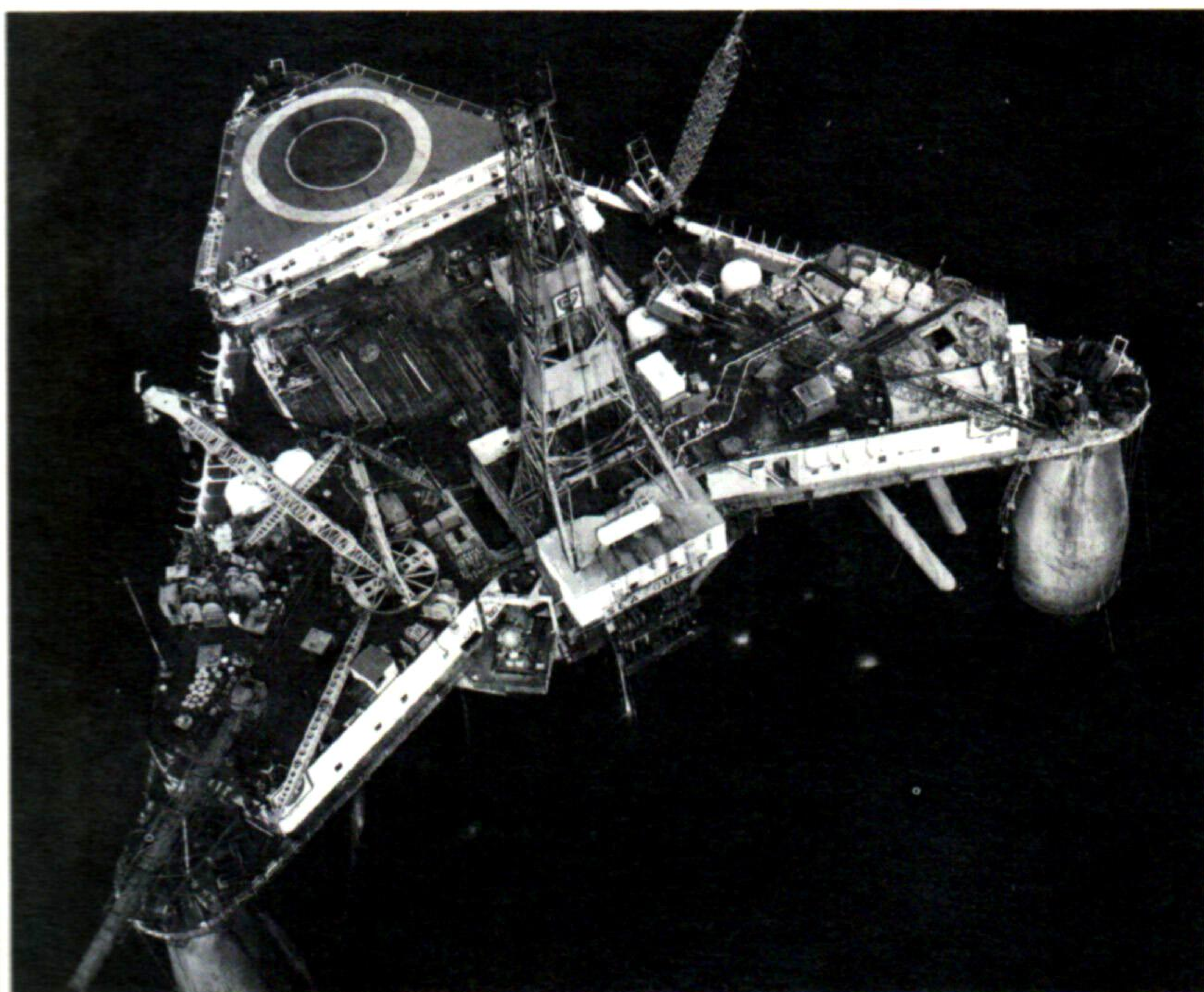
Para simular la economía de este país, los economistas utilizan un "modelo", no un modelo físico, sino más bien un diagrama complejo que ilustra cómo interactúan entre sí todos los componentes de la economía. En los programas *1984* el modelo es muy simple, dado que apenas si contiene cinco secciones: el gobierno, los bancos, la población, la industria y el resto del mundo.

El Ministerio de Hacienda también posee un modelo económico, que el gobierno británico utiliza para hacer diversas predicciones acerca de la economía nacional. Como cabe imaginar, este modelo es algo más complicado. El programa de simulación, que contiene el modelo, se denomina *AMODEL* y se ejecuta en un ordenador central Sperry 1100. Para efectuar sus predicciones emplea 1 100 variables económicas distintas (la tasa de inflación, desempleo, etc.) reunidas durante los últimos 10 años. Sólo estos datos ocupan alrededor de 250 Kbytes de memoria, pero cuando se ejecuta el programa al máximo de su capacidad se utilizan cerca de ocho Mbytes.

Típicamente, el modelo se puede emplear para predecir cifras de desempleo, el nivel de importaciones y exportaciones, el valor al cambio del dólar y muchos otros índices económicos. Para que se haga una idea de la complejidad del modelo, basta con decir que predecir la tasa de inflación para el año que viene llevaría dos o tres minutos de procesador. No parece mucho tiempo, pero hay que tener presente que el procesador ejecuta alrededor de 10 millones de instrucciones en código máquina en apenas un segundo. Imprimir la predicción lleva otros 15 o 20 minutos.

Los juegos de simulación por ordenador le ofrecen la oportunidad de dirigir su propio negocio (o incluso la economía de toda una nación) sin correr el riesgo de perder todos los ahorros de su vida si la empresa se declara en quiebra. Lamentablemente, ¡uno tampoco puede disfrutar de los posibles beneficios!

FINANCIAL MICRO TIMES



Associated Press

ORO NEGRO

"Dallas" es un juego de gestión que coloca al jugador al frente de su propia compañía petrolífera. Comenzando con $100 millones en efectivo, se deben acumular $200 millones con el fin de adquirir a su mayor rival, Euing Associates. (El error ortográfico en el apellido familiar de la popular serie de televisión es intencional.)

En la pantalla hay un mapa del estado de Texas y a medida que avanza el juego se tiene la oportunidad de ir ganando "concesiones" en cada uno de los cuadrados. Para obtener la concesión es necesario hacer una oferta con éxito, pero una vez que se ha efectuado una, hay que perforar un pozo de prueba para buscar el petróleo. Si descubre petróleo en su campo, entonces instala un equipo de perforación y medios de producción, y finalmente un oleoducto. Si se compromete demasiado (comprando, p. ej., muchas concesiones antes de disponer de ingresos por petróleo), entonces tal vez se vea obligado a acudir a un banco para obtener un crédito, con lo cual empieza a pagar intereses. Si el préstamo supera la cantidad de $20 millones, puede ser absorbido por Euing Associates.

Este programa lo vende Cases Computer Simulations para una serie de micros personales, entre ellos el BBC Micro, el Spectrum, el Oric y el Electron.

ALTAS FINANZAS

En "Airline" usted es el presidente de L-AIR, unas líneas aéreas privadas cuyo capital, al comienzo del juego, es de £3 millones. En un período de siete años debe incrementar el activo a 30 millones, punto en el que está en condiciones de adquirir la British Airways y ganar el juego.

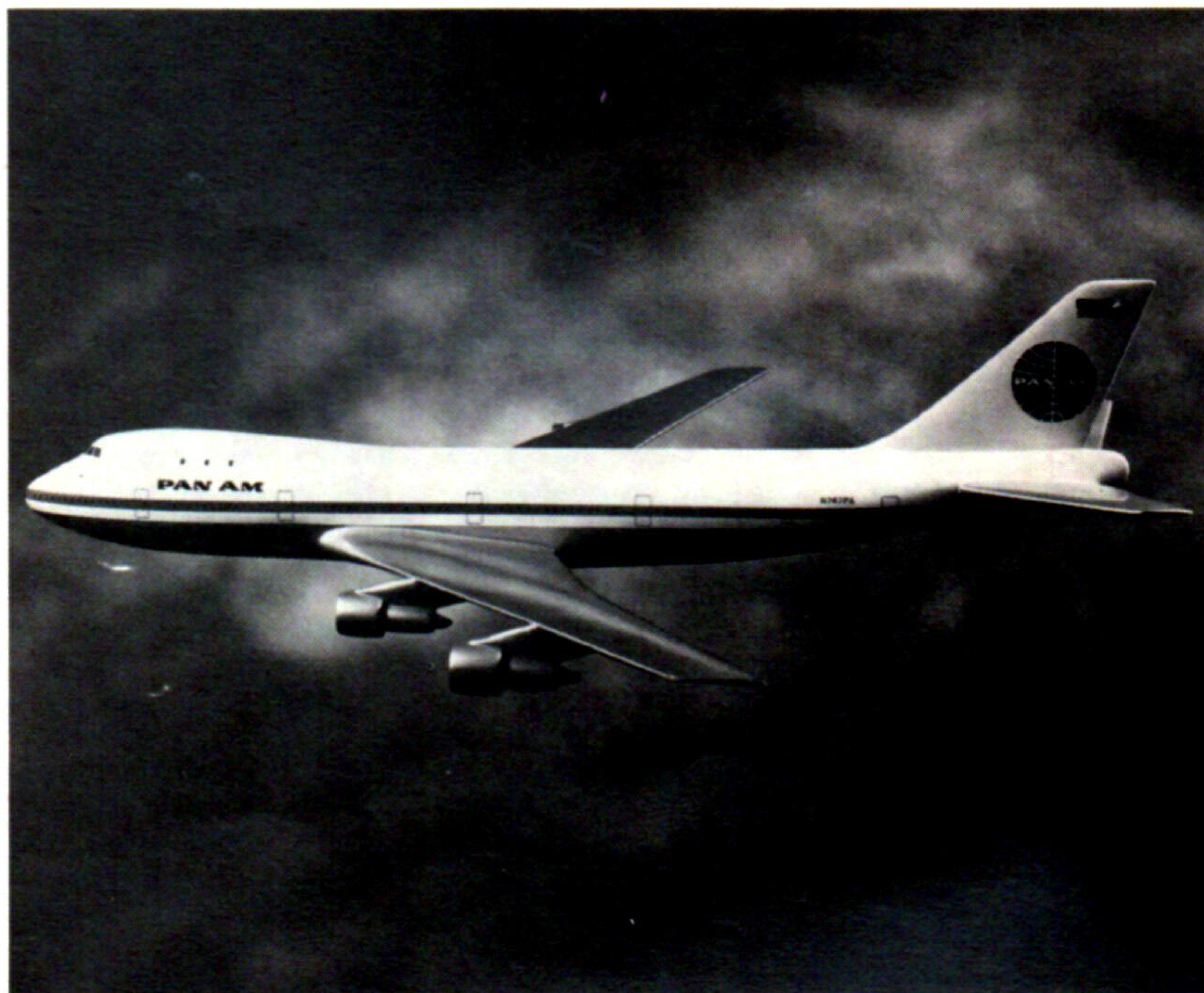
Al principio de cada año financiero hay que decidir con cuántos aviones operar, utilizando estimaciones que predicen el número de pasajeros. Inicialmente no se dispone de suficiente dinero para comprar un avión (valen £10 millones cada uno), de modo que hay que alquilarlos. En años más rentables el jugador habrá de decidir si es mejor alquilar un avión o comprarlo, y esto depende tanto de las tarifas de alquiler como de los tipos de interés.

Se debe decidir, asimismo, el nivel de tripulación y mantenimiento de su avión. Si es demasiado bajo, entonces quizá no se disponga de personal suficiente o de suficientes aviones, y se podría llegar a cancelar algunos vuelos. Si es demasiado alto, se estará derrochando dinero.

En ocasiones en la pantalla aparece una "cinta de teleimpresora" que contiene numerosos mensajes de télex. Por ejemplo, quizá la OPEP haya aumentado el precio de los crudos (y, por consiguiente, el combustible) o al jugador se le ha concedido licencia para una nueva ruta aérea.

Al final del año una hoja de balance proporciona el rendimiento global de las L-AIR y un informe del gerente acerca del éxito obtenido por la empresa. Si en el primer año se pierden £10 millones, la empresa se liquida.

El juego está a la venta para el Spectrum, Oric, Electron y BBC Micro, y lo produce Cases Computer Simulations.



Associated Press



Una oportunidad de primera



Associated Press

En este juego usted es el primer ministro de Gran Bretaña en el año 1984. Su objetivo es el de mantenerse en su cargo el mayor tiempo posible, y su popularidad está determinada por el éxito que consiga al equilibrar las cuentas del país.

Al comienzo de su mandato tiene que tomar diversas decisiones en materia financiera. A modo de ayuda dispone de tres indicadores económicos, incluyendo las tasas de inflación y desempleo, etc. Al inicio de cada año un gráfico indica cómo ha cambiado cada indicador durante sus años de gestión.

Su primera decisión consiste en determinar el tipo mínimo de crédito, que determina el tipo de interés para el año. Usted tiene luego que negociar los incrementos salariales para la Administración Pública, el sector público y el sector privado.

Después de los convenios salariales, debe decidir acerca de los niveles de gastos de los diversos departamentos y ministerios del gobierno. Por último, puede anunciar su presupuesto, que le da la oportunidad de recaudar dinero a base de impuestos. Al final de cada año de su mandato de cinco años, un sondeo de opinión refleja la popularidad de su política y contribuirá al objetivo último que usted persigue, el de ser reelegido al final del juego.

Incentive Software produce el juego *1984* para los modelos BBC Micro y el Spectrum.

COSECHA DORADA

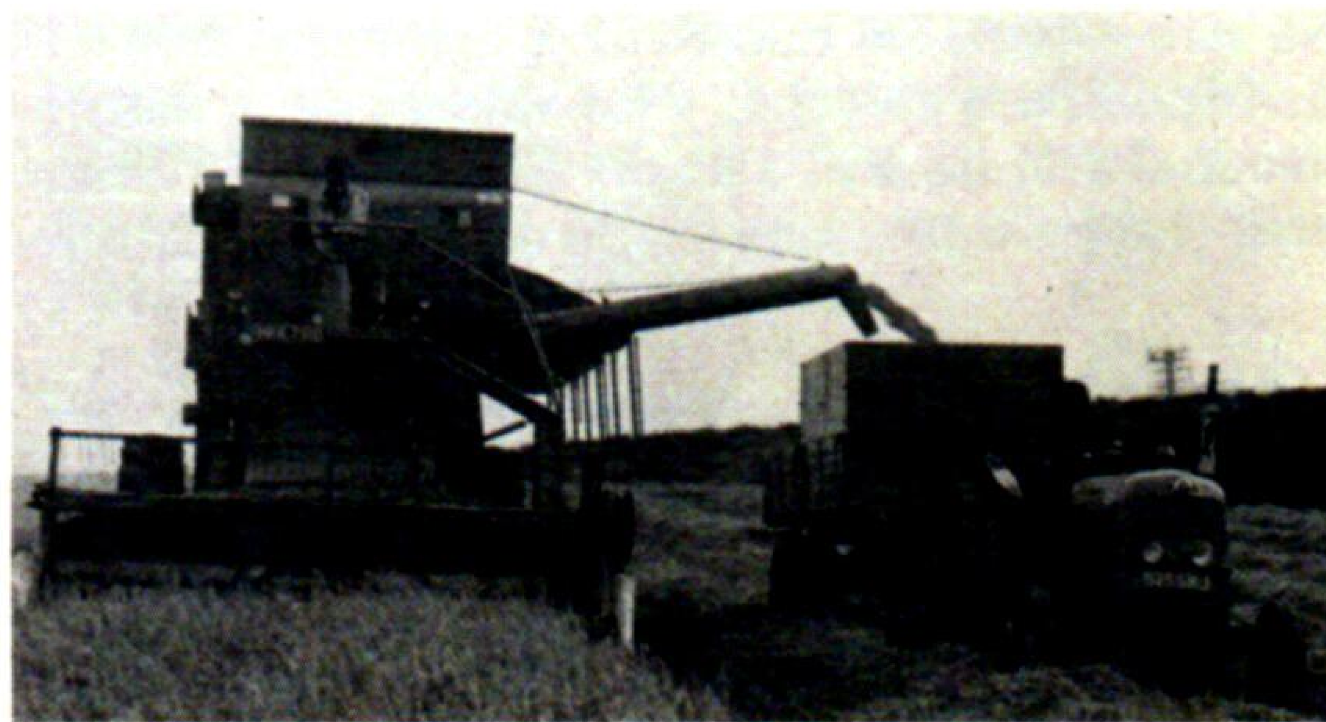
"Corn cropper", de Cases Computer Simulations, disponible para el Spectrum, el Electron o el BBC Micro, simula la administración de una gran granja de trigo. Se empieza a jugar con £50 000 y se tiene como objetivo dirigir la granja de modo que al cabo de cinco años posea un activo total de £250 000.

Al principio el participante apenas posee 30 acres de tierra que se han de sembrar. Para hacerlo necesita comprar semilla, alquilar un tractor y pagar el riego en caso de que no llueva lo suficiente. Al principio de cada mes el jugador dispone de las previsiones de precipitaciones y temperatura y decide, en consecuencia, si sembrar o no.

A medida que van transcurriendo los meses tiene que estar atento al "informe del estado de la cosecha" (un calendario que dice si hay trigo listo para su cosecha).

Pueden aparecer otros imponderables: las ratas se podrían comer parte de sus semillas; si no ha pulverizado el grano, los insectos lo atacarán, y la helada puede producir daños considerables.

A medida que avanza el juego, usted intenta obtener unos ingresos constantes producto de la venta del trigo cosechado, de modo tal que pueda reinvertir los ingresos en más tierras y más semillas. Es bastante difícil acrecentar su activo por encima de las £100 000, en especial porque no se le permite girar en descubierto al banco. Como último recurso, siempre puede vender parte de sus tierras para hacerse con algo más de dinero.



Associated Press



Dirección desde la butaca
Estos juegos de gestión por ordenador personal utilizan técnicas de simulación para poner al jugador al mando de unas líneas aéreas, una granja de trigo, una compañía petrolífera y la economía británica. El grado de realismo varía de acuerdo a la cantidad de factores de control que admita el juego y a la complejidad de los modelos de interacción de factores del juego

Ian McKinnell

Radio de acción

Vamos a analizar el trazado de líneas, el reloj interno y los demás procedimientos necesarios para el escenario de nuestro juego

En el primer capítulo definimos una superficie de la pantalla del BBC en modalidad 5 como el "campo de minas" en el cual se habrá de desarrollar nuestro juego. Definimos las formas de las minas, del detector y del ayudante, desarrollamos procedimientos para colocar un cierto número de minas al azar en el campo de minas, y situamos el detector y el ayudante en sus posiciones de comienzo. Para hacer que la visualización sea más atractiva podemos dibujar un borde alrededor de la superficie del campo de minas. La forma más sencilla de hacerlo consiste en utilizar las instrucciones en alta resolución MOVE y DRAW.

En el BBC/Electron existe un problema al mezclar gráficos en alta resolución con caracteres en baja resolución: las visualizaciones diferentes utilizan sistemas de coordenadas diferentes. Ya hemos analizado en detalle la visualización en baja resolución cuando la aplicamos para posicionar las minas y los caracteres con la instrucción PRINT TAB(X,Y). En este sistema, el origen (el punto donde tanto X como Y son cero) es el rincón superior izquierdo. En este sistema los valores de X aumentan de 0 a 19 de izquierda a derecha, y los valores de Y aumentan de 0 a 31 de arriba abajo. La modalidad 2 también emplea una visualización de 20 por 32 caracteres, pero el resto de modalidades muestra una cantidad distinta de caracteres y, por lo tanto, utilizan distintas coordenadas para cada carácter.

Las ocho modalidades de visualización del BBC/Electron proporcionan tres resoluciones distintas (640 por 256, 320 por 256 y 160 por 256) y, aun así, todas ellas emplean el mismo sistema de coordenadas. Aunque al principio este sistema resulta algo confuso, es una verdadera ayuda para la programación, porque los gráficos diseñados para una modalidad se podrán usar para las pantallas de otras modalidades.

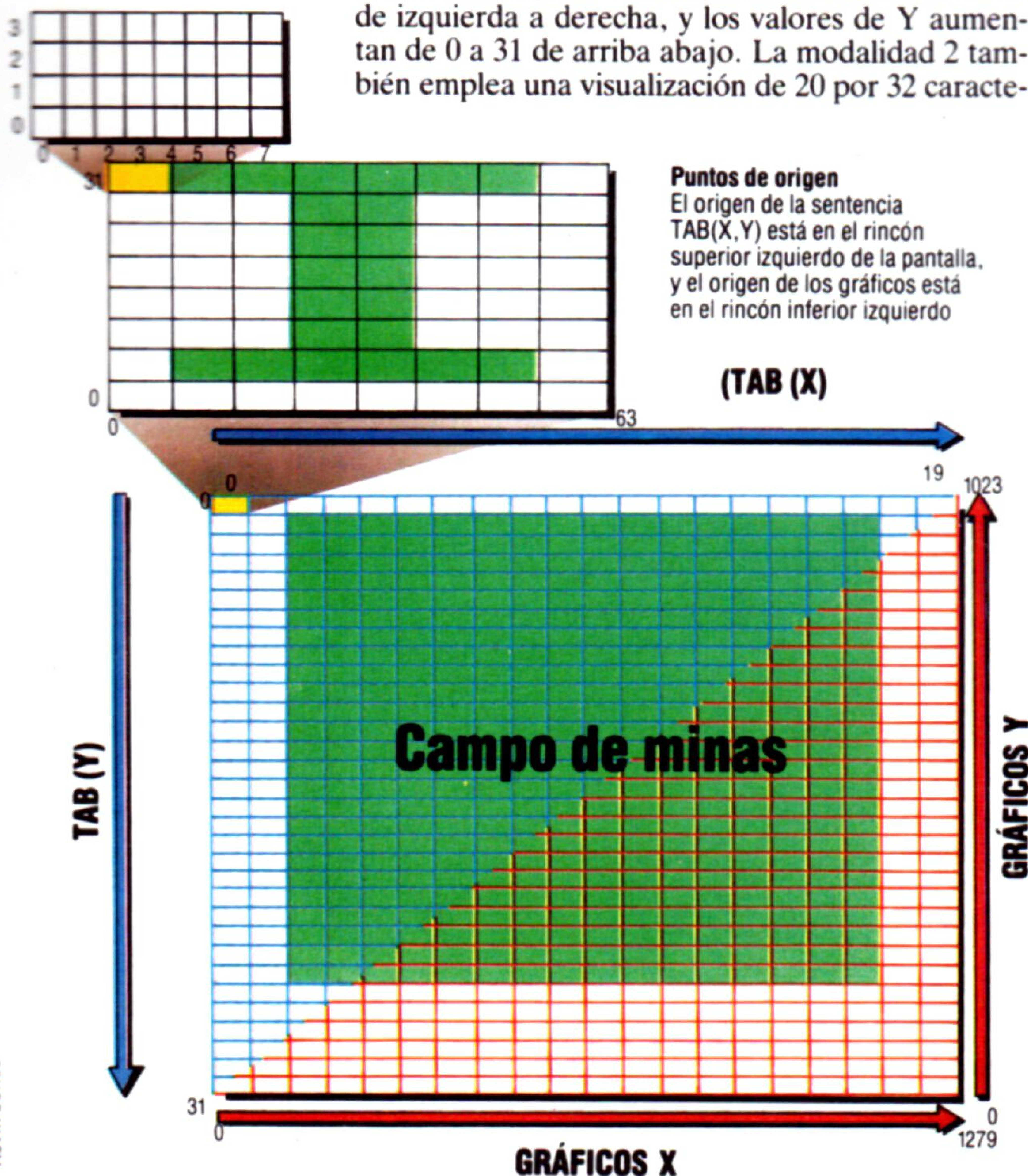
El sistema de coordenadas del BBC/Electron trata la pantalla como si tuviera una resolución de 1280 por 1024. Ésta, por supuesto, es una visualización más grande de la que pueden producir ambas máquinas, ¡evidentemente para permitir futuros desarrollos del BBC! Todas las coordenadas se dan como números comprendidos en la escala entre 0 y 1279 a lo ancho de la pantalla y entre 0 y 1023 a lo alto de la misma. El BASIC BBC los convierte automáticamente en los valores apropiados para cualquiera de las tres resoluciones que se esté utilizando.

Mezclar gráficos con caracteres resulta un poco más difícil, porque el origen de coordenadas de los gráficos está situado en el rincón inferior izquierdo de la pantalla, en vez de en el rincón superior izquierdo empleado en el sistema de coordenadas de caracteres.

En la modalidad 5, la resolución de 160 por 256 pixels guarda una relación directa con el número de caracteres que se pueden mostrar. Cada uno de los caracteres se forma a partir de una cuadrícula de ocho por ocho pixels. Dado que a lo ancho de la pantalla hay sitio para 20 caracteres, ello significa que a lo ancho debe haber $8 \times 20 = 160$ pixels. Del mismo modo, a lo alto de la pantalla debe haber $32 \times 8 = 256$ pixels. Para relacionar esto con el sistema de coordenadas en alta resolución debemos recordar que un pixel es la menor superficie de luz de la pantalla que se puede encender o apagar individualmente. En el sistema de alta resolución, en la dirección X hay 1280 coordenadas diferentes. Si dividimos esta cantidad por el número de pixels de la dirección X, obtenemos $1280/160 = 8$. Del mismo modo, en la dirección Y, dividiendo las coordenadas de alta resolución por el número de pixels obtenemos $1024/256 = 4$. Esto significa que cada pixel se puede encender haciendo referencia a cualquiera de las diversas posiciones del sistema de coordenadas de 1280 por 1024. La ilustración refleja cómo se puede utilizar una gama de coordenadas para encender (o apagar) un pixel. Trazando (7,3) se iluminaría el mismo pixel que trazando (0,0) o (5,2), y así sucesivamente.

Podemos utilizar este hecho para relacionar posiciones de caracteres con coordenadas de alta reso-

En un cuadrado
Los caracteres se definen en una matriz de ocho pixels por ocho. En la modalidad 5, en realidad ésta es rectangular, siendo de 64 puntos de alta resolución de ancho por 32 puntos de alto, de modo que cada pixel de caracteres de la modalidad 5 debe medir 8 puntos de alta resolución de ancho por 4 puntos de alto. Trazar (PLOT) o dibujar (DRAW) cualquiera de los puntos de un pixel hará que se ilumine todo el pixel: PLOT 7,3, por consiguiente, equivale a PLOT 5,2



Puntos de origen
El origen de la sentencia TAB(X,Y) está en el rincón superior izquierdo de la pantalla, y el origen de los gráficos está en el rincón inferior izquierdo



lución. En el eje horizontal, si un pixel equivale a ocho unidades, la anchura de un carácter equivaldrá a 64 unidades. Cuatro unidades equivalen a un pixel en la dirección vertical, lo que implica que la altura de cada carácter es de 32 unidades. Las fronteras de la pantalla se pueden calcular ahora en términos de coordenadas en alta resolución para las instrucciones MOVE y DRAW. La ilustración del trazado de pantalla muestra cuáles son estas fronteras.

Ahora podemos calcular las coordenadas del rincón inferior izquierdo y del rincón superior derecho del campo de minas (todas las otras coordenadas para el borde se obtienen a partir de estos dos puntos). Como podemos ver a partir del diagrama, las coordenadas del rincón inferior del borde son (120,188). Las coordenadas del rincón superior derecho son (1152,992).

El siguiente procedimiento dibuja un borde alrededor del margen de la superficie definida. GCOL 0,1 establece el color lógico que se utilizará para los gráficos. El primer número define el tipo de trazado, que examinaremos más adelante, y el segundo define el color. Las instrucciones MOVE desplazan el cursor para gráficos (sin dibujar) desde el origen hasta el rincón inferior izquierdo del borde. Las instrucciones DRAW que le siguen dibujan líneas rectas desde la posición en curso de la pantalla hasta el punto especificado.

```
2470 DEF PROCdibujar-borde
2480 GCOL 0,1
2490 MOVE 120,188
2500 DRAW 120,992
2510 DRAW 1152,992
2520 DRAW 1152,188
2530 DRAW 120,188
2540 ENDPROC
```

El reloj interno

El BBC y el Electron poseen un reloj interno al cual se puede acceder fácilmente desde el BASIC utilizando la variable reservada TIME. Cuando se solicita que imprima el valor de TIME, el ordenador devuelve un número que corresponde al tiempo, expresado en centésimas de segundo, dado que a la variable se le dio la última vez el valor de cero. El procedimiento "establecer-tiempo" imprime la palabra "Tiempo", su valor de comienzo y pone el valor cero a la variable TIME. A este procedimiento se lo llama durante la rutina de preparación e inicia el reloj para el juego.

```
2640 DEF PROCestablecer-tiempo
2650 PRINTTAB(2,27)"Tiempo    02:00"
2660 TIME = 0
2670 ENDPROC
```

Dentro del bucle principal del programa, se debe actualizar el tiempo visualizado en la pantalla. Visualizar el tiempo en segundos sería muy directo; simplemente dividiríamos la variable TIME por 100, para convertir su valor a segundos, imprimiríamos este valor en la pantalla, y así sucesivamente. Sin embargo, es posible convertir TIME a minutos y segundos haciendo uso de las instrucciones del BASIC BBC DIV y MOD. TIME DIV 100 devolvería el número de segundos como un número entero; (TIME DIV 100)MOD 60 contaría los segundos de 0 a 59 y luego volvería a comenzar desde cero. Ello se debe a que la instrucción MOD 60 da el valor del resto después de la división por 60. De modo que, por ejemplo, 63/60 es 1, con un resto de 3. (63/60)MOD 60, por consiguiente, sería 3. Los minutos se pueden aislar de la misma manera y visualizar mediante la utilización de (TIME DIV 6000)MOD 60.

Éste es el procedimiento que se aplicará para actualizar el tiempo durante el juego:

```
2900 DEF PROCactualizar-tiempo
2910 seg$ = STR$(((12100-TIME) DIV 100)MOD 60)
2920 min$ = STR$(((12100-TIME) DIV 6000)MOD 60)
2930 REM ** sumar ceros izquierda**
2940 seg$ = LEFT$(cero$,2-LEN(seg$)) + seg$
2950 min$ = LEFT$(cero$,2-LEN(min$)) + min$
2960 time$ = min$ + ":" + seg$
2970 PRINTTAB(11,27);time$
2980 ENDPROC
```

Como puede apreciar a partir de este procedimiento, hemos ido un paso más hacia adelante. Además de estar dividido en minutos y segundos, el tiempo en realidad se contará hacia atrás desde dos minutos a cero. También se incluye una breve rutina de manipulación de series para asegurar que las visualizaciones para los segundos y los minutos siempre tendrán dos dígitos, agregando ceros a la izquierda cuando ello sea necesario.

Para completar la preparación del juego se requieren aún otros dos cortos procedimientos. Durante el juego el participante tiene cuatro vidas; por consiguiente, necesitamos visualizar, en la parte inferior de la pantalla, la cantidad de vidas que le quedan. Al principio éstas serán tres, visualizadas en forma de tres caracteres de "ayudante", tal como lo habíamos definido en el capítulo anterior (véase p. 873). Se empleará un "contador" variable para determinar la cantidad de vidas utilizadas. Inicialmente, éste valdrá uno.

```
2690 DEF PROCestablecer-hombres
2700 hombres$ = CHR$(226) + CHR$(226) + CHR$(226)
2710 contador = 1
2720 COLOUR 1
2730 PRINTTAB(2,30);hombres$
2740 COLOUR 2
2750 ENDPROC
```

La rutina final de preparación inicializa los marcadores y los visualiza en la pantalla. El valor de "maxmarcador$" no se inicializa en este procedimiento. Por tanto, estableceremos su valor inicial sólo al comienzo del programa.

```
2770 DEF PROCestablecer-marcador
2780 marcador = 0:marcador$ = "00000"
2790 PRINTTAB(2,28)"Marcador       00000"
2800 PRINTTAB(2,29)"Max marcador  ";max-marcador$
2810 ENDPROC
```

Ahora que ya hemos ensamblado todos los procedimientos de la parte de preparación del programa, podemos construir un procedimiento de mayor nivel para llamarlos a todos ellos. En el último capítulo habíamos llamado a todos los procedimientos que habíamos ensamblado directamente desde un corto programa principal. Ahora usted debe eliminar aquellas líneas y agregar las siguientes:

```
1880 DEF PROCpreparacion
1890 COLOUR 2
1900 bandera-final = 0
1910 PROCinicializar-variables
1920 PROCdefinir-caracteres
1940 PROCcolocar-minas(40)
1950 PROCdibujar-borde
1960 PROCestablecer-tiempo
1970 PROCestablecer-marcador
1980 PROCestablecer-hombres
1990 PROCsituar-sujetos
2000 ENDPROC
```

Estamos ahora en la etapa en la que podemos escribir un corto programa principal para llamar al procedimiento "preparación" y actualizar el tiempo en un bucle REPEAT...UNTIL. Añada a su programa estas líneas:

```
10 REM****PROGRAMA DE LLAMADA****
20 maxmarcador$ = "00000"
30 PROCpreparacion
40 REPEAT
50 PROCactualizar-tiempo
60 UNTIL TIME > 12099
70 END
```

Ejecutante versátil

La Brother EP-44 puede funcionar como impresora y también ser una eficiente calculadora y terminal de comunicaciones

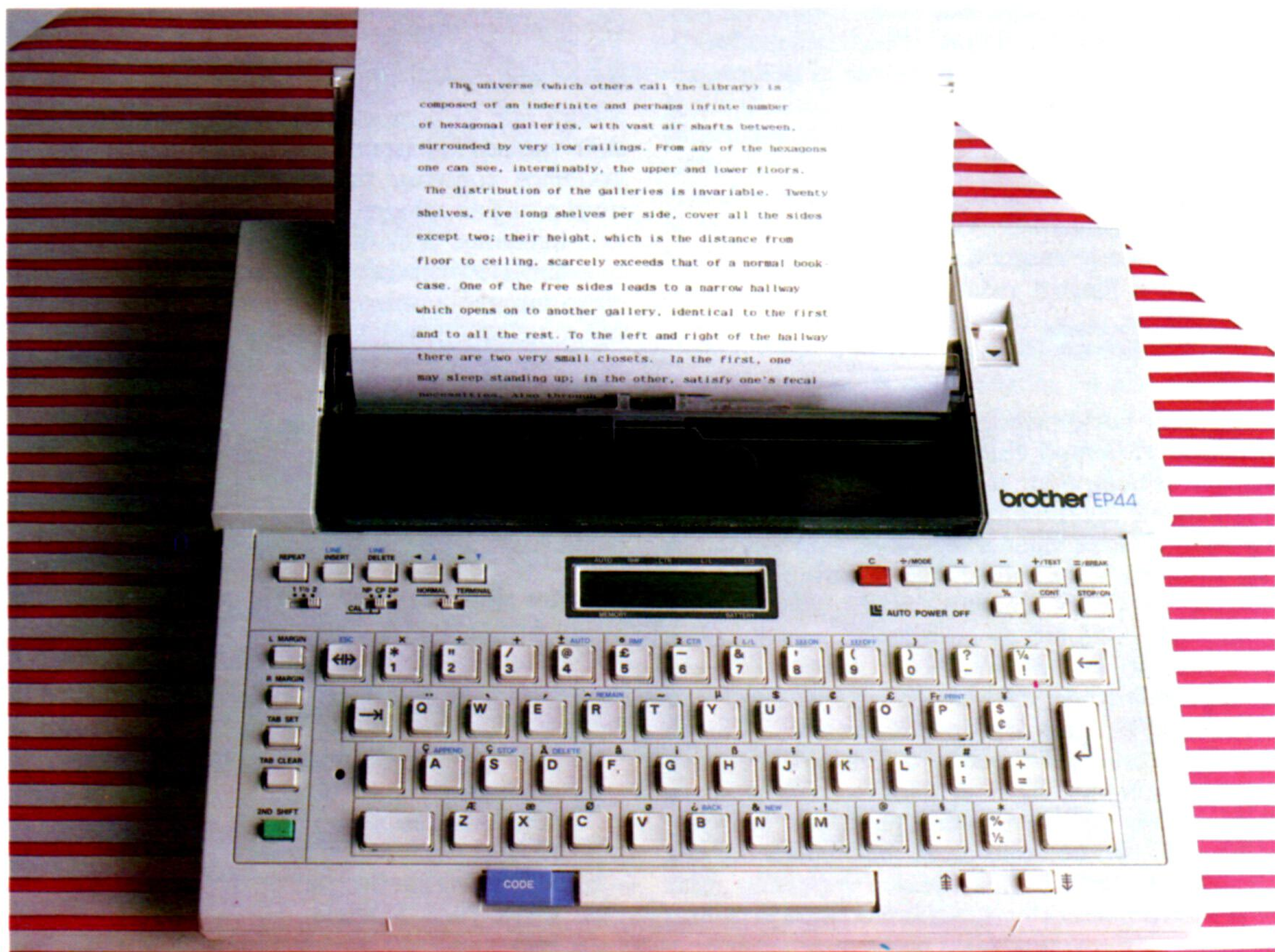
La Brother EP-44 pesa 2,2 k y funciona a pilas, por lo que es una máquina auténticamente portátil; se le puede agregar como accesorio un transformador de corriente. En el teclado, además de las teclas convencionales de máquina de escribir, hay cuatro teclas para tratamiento de textos y siete teclas de calculadora. Pero la característica más notable es la visualización en cristal líquido (LCD) de 15 caracteres. Ésta permite que el usuario visualice y edite el texto antes de imprimirlo.

Para utilizar la EP-44 como máquina de escribir convencional, el interruptor Print Mode Selector (selector de modalidad de impresión) se coloca en "Direct Print" (impresión directa). En esta modalidad, el texto aparece en la "ventana" de LCD y, simultáneamente, se imprime en papel. Las tabulaciones, el espaciado de las líneas y los márgenes se fijan del mismo modo que en una máquina de escribir convencional. Sin embargo, el sistema de alimentación de papel es poco frecuente: en vez de mover manualmente el rodillo portapapel, el usuario debe pulsar uno de dos botones para desplazar el papel hacia arriba o hacia abajo. El teclado no responde del todo al estándar de una máquina de escribir electrónica, pero es perfectamente adecuado.

La EP-44 utiliza un cabezal de impresión térmico para "quemar" una serie de puntos sobre el papel y formar así el carácter deseado. La mayoría de las impresoras térmicas emplean una matriz de puntos de nueve por siete para conformar los caracteres, pero con la matriz de la EP-44, de 24 por 18, se obtiene una mejor calidad de impresión. Otra de las impresoras de la gama Brother, la EP-22, utiliza menos puntos y produce una impresión notablemente más pobre.

Una importante desventaja del sistema de impresión térmico es la baja velocidad de la impresión que, en este caso, está por debajo de los 16 caracteres por segundo. Cuando se emplea la Brother como máquina de escribir esto no es un inconveniente serio, pero cuando se utiliza la máquina como impresora de ordenador la escasa velocidad se hace muy notoria; una buena impresora matricial imprimirá el texto a una velocidad 10 veces mayor que la Brother. El otro problema de este sistema es el papel térmico especial que requiere; éste es caro y su aspecto es muy satinado.

Si se coloca el interruptor Print Mode Selector en "Correction Print" (corregir impresión), la ventana LCD demuestra su verdadera utilidad. En esta modalidad el texto aparece en la ventana pero se produce una demora antes de que sea impreso. De hecho, el texto impreso en papel lleva 15 caracteres



El teclado
Las 51 teclas de impresión están ligeramente esculpidas, son de presión suave y responden a un trazado QWERTY no estándar. Hay dos teclas de cambio de caracter con bloqueo, una de cambio de símbolo y una de cambio de función. Las teclas de caracteres son teclas mulfifunción, que ofrecen una gama de caracteres extranjeros y acentos ortográficos. Hay 21 teclas de función que controlan las modalidades de impresión, de calculadora y de memoria, la colocación de márgenes, tabuladores y el movimiento del papel



de atraso con respecto a la letra que se está digitando en cada momento. Esto produce al principio una sensación extraña, pero permite la alteración de cualquiera de los últimos 15 caracteres antes de que sean impresos, lo que resulta útil si su mecanografía no es muy precisa y, por cierto, representa una gran mejora sobre los líquidos correctores. La edición se realiza en la visualización de LCD mediante la utilización de dos teclas para control del cursor, para seleccionar el carácter a modificar. Hay, asimismo, una modalidad "Line-By-Line" (línea a línea), en la cual la EP-44, antes de escribir una línea completa de texto, espera a que se pulse la tecla Return. Ello posibilita la edición de una línea entera en vez de sólo los 15 últimos caracteres mecanografiados.

Los usuarios de tratamiento de textos se acostumbran enseguida al hecho de que el texto se formatea automáticamente y pierden pronto el hábito de pulsar la tecla Return al final de cada línea de texto. La EP-44 posee una modalidad "Auto Return" (retorno automático), en la cual se inicia automáticamente una nueva línea si se pulsa la barra espaciadora en alguna de las seis últimas posiciones de una línea.

La calculadora incorporada de la EP-44 permite realizar cálculos simples en la visualización LCD sin interrumpir la impresión. Por ejemplo, si se está entrando una factura de venta y se necesita calcular el ITE o los descuentos usuales, puede utilizarse la calculadora para obtener las cifras en cuestión y continuar imprimiendo los detalles de la factura.

Todas las características que hemos mencionado aquí también se pueden encontrar en muchas máquinas de escribir electrónicas modernas. Pero la EP-44 también se puede emplear como un sencillo procesador de textos. Esta característica utiliza los 3,5 Kbytes de memoria interna que emplea la Brother para almacenar texto. Esta memoria puede resultar pequeña si se la compara con aquella de que disponen la mayoría de los ordenadores personales, pero es suficiente para producir una carta de tres páginas. A las instrucciones necesarias para el tratamiento de textos se accede utilizando las teclas normales de máquina de escribir conjuntamente con una tecla "Code" (código) especial de color azul. Por consiguiente, para introducir texto en la memoria lo único que hay que hacer es pulsar la tecla Code y la letra "N" (de *new text*: texto nuevo). Se puede entrar el texto tanto en las modalidades Direct como Correction Print.

Después de haber entrado el texto en la memoria, para editarlo se utilizan la visualización LCD y las teclas del cursor. Las teclas del cursor permiten desplazar la ventana LCD hacia arriba, hacia abajo, hacia la izquierda y la derecha, de modo que se puede contemplar el texto completo e insertar, eliminar o corregir caracteres a voluntad. La única limitación reside en el hecho de que las palabras no se pueden desplazar hasta la línea de abajo si la línea que se está editando es demasiado larga, lo que significa a veces la imposibilidad de reformatear un documento.

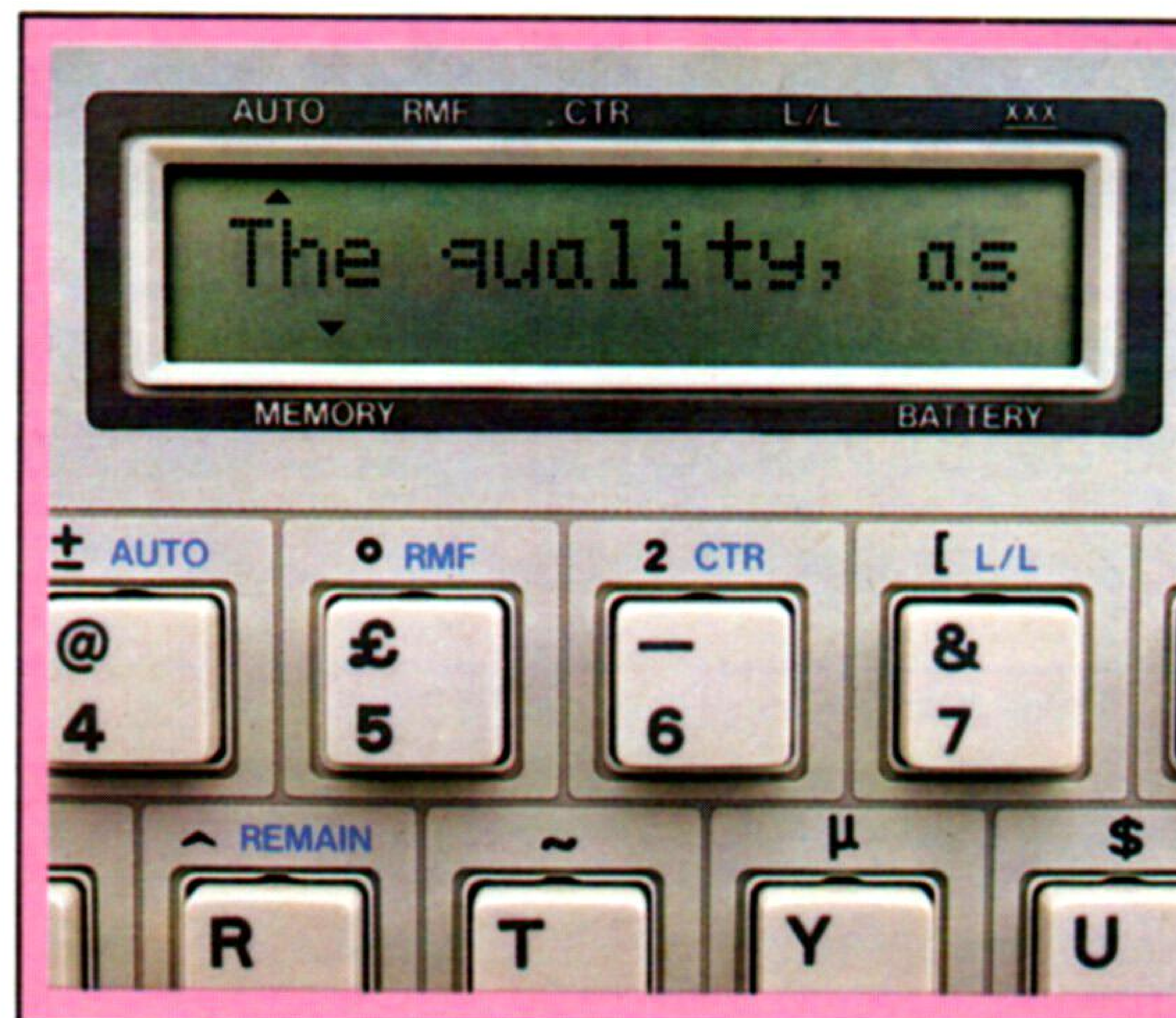
La gran ventaja de entrar el texto de esta manera es que éste se puede alterar sin necesidad de volver a mecanografiar todo el documento entero. Por ejemplo, a menudo hay que enviar la misma carta a varias personas diferentes; para hacerlo, simplemente se edita el texto en la memoria para insertar el nombre y la dirección correspondiente, y luego se pulsa la tecla Code junto con la letra "P" (de *print text*: imprimir texto). Entonces se imprimirá su carta modificada.

Quitando una pequeña cubierta a uno de los lados de la EP-44 queda al descubierto un conector en serie RS232. Un interruptor señalado como "Terminal" convierte a la Brother en una impresora para ordenador, si bien es necesario un poco de práctica para establecer el valor correcto de la velocidad en baudios, longitud de las palabras, etc., para su máquina en particular. Adaptar la EP-44 es sumamente fácil: la visualización LCD va pasando las diversas velocidades y otros detalles, y usted tan sólo pulsa el interruptor Mode cuando los detalles expuestos son los específicos para su máquina.

Lamentablemente, la Brother no puede manipular el papel de bobina habitual y se la ha de alimentar de forma manual con hojas sueltas. Pero la calidad de impresión es mejor que la de una impresora matricial, por lo que la EP-44 es ideal para la producción de cartas y documentos cortos.

El hecho de que el interruptor se señale como "Terminal" y no como "Printer" (impresora), refleja que la EP-44 puede enviar datos, además de recibirlos. Esto significa que la Brother se puede conectar a un modem, permitiendo que el usuario se comunique, por vía telefónica, con otros usuarios de ordenadores personales o con máquinas más grandes que utilicen los estándares de comunicación apropiados.

En conjunto, la Brother EP-44 destaca por su versatilidad y por el hecho de que, siendo una máquina pequeña, reúna tantas funciones diferentes.



LCD, ¿OK?

La pantalla LCD de la EP-44 visualiza 15 caracteres, cinco indicadores de modalidad de impresión y una alerta del estado de las pilas. El texto se puede editar en la pantalla, permitiendo, por tanto, un tratamiento de textos sencillo. La intensidad de visualización de los caracteres se puede ajustar de acuerdo a las condiciones de iluminación ambiental

Ian McKinnell

Salida impresa

Con la cinta colocada, se puede utilizar cualquier papel de maquina de acabado satinado, pero la alimentacion del papel no acepta papel carbon. La calidad de impresion es buena con ambas clases de papel.
El texto se puede:
Centrar & subrayar
hay caracteres extranjeros (ÆØ¿¢µπ),
subindices y exponentes ($X^2 = N_2 + T_0$).

Contador alfabético

El test en cascada se puede utilizar no sólo en una comparación de datos numéricos, sino también alfanuméricos

INICIO
ENTRAR UNA FRASE
FRASE
1
TOMAR UNA LETRA
ES LA A? — SÍ — SUMAR 1 A CONTADOR 1
NO
ES LA E? — SÍ — SUMAR 1 A CONTADOR 2
NO
ES LA I? — SÍ — SUMAR 1 A CONTADOR 3
NO
ES LA O? — SÍ — SUMAR 1 A CONTADOR 4
NO
ES LA U? — SÍ — SUMAR 1 A CONTADOR 5
NO
ESPACIO BLANCO? — SÍ — SUMAR 1 A CONTADOR 6
NO
SUMAR 1 A CONTADOR 7
QUEDAN MÁS LETRAS? — SÍ — 1
NO
ESCRIBIR VALOR DE LOS CONTADORES
FIN

Todos los supuestos vistos hasta el momento en los que se ha empleado el test en cascada se han basado en comparaciones con cifras. Ahora bien, esta técnica de diagramación tiene efectividad tanto si se emplea la comparación de datos numéricos como de datos alfanuméricos, tal como se muestra en el siguiente ejemplo. En él, basándose en la introducción de una frase cualquiera entrada por teclado, se pretende conseguir como final lógico una serie de resultados basados en el número de veces que un carácter esté contenido en la frase. Así, se empleará un contador por cada vocal, uno que dé el total de consonantes existentes y otro que lleve el total de espacios en blanco contenidos. Con la visualización de estos siete campos se obtendrá la solución del programa.

Veamos a continuación un ejemplo introduciendo: "LA PELOTA SE ROMPIO".

Nos devolverá el siguiente resultado:

CARÁCTER	TOTAL
A	2
E	2
I	1
O	3
U	0
CONSONANTES	8
ESPACIOS	3

Puede apreciarse que tras las preguntas por cada una de las vocales, se ha inquirido por el espacio en blanco, con lo que, por eliminación, tras el paso por estas seis comparaciones previas, se ha tomado, por defecto, el carácter descendente como una consonante.

Consideraciones generales

El cuerpo del ordinograma podría haberse ampliado sustancialmente en caso de haberse deseado una mayor depuración del resultado, por ejemplo, verificando la inclusión de algún número, o incluso la utilización de algún carácter especial dentro de la frase de nuestro ejemplo.

Con lo que se llega, en un último supuesto, al control total, obtenido tras el empleo de un contador para cada una de las diferentes consonantes de nuestro alfabeto.

Sea cual fuere la opción escogida, obsérvese que su desarrollo no ganaría en complejidad, ya que se trataría de repetir unas mismas fases con la única variación del nombre del carácter y del número del contador correspondiente. Es decir, que la dificultad sería lo laborioso de plasmar el desarrollo del planteamiento, conforme se utilice mayor número de filtros de decisión.



Un brazo eficiente

Los trazadores digitales acaban con la laboriosa tarea de transferir una imagen o un diseño desde el papel al ordenador

Un trazador digital es un instrumento sencillo que permite copiar un dibujo, una fotografía o un diseño a la pantalla del ordenador mediante la técnica de reseguir todos sus trazos con un brazo articulado. La facilidad con que esto se lleva a cabo depende en gran parte del software que acompaña al equipo. Aquí vamos a analizar cuatro trazadores digitales, tres para el BBC Micro y uno para el Sinclair Spectrum.

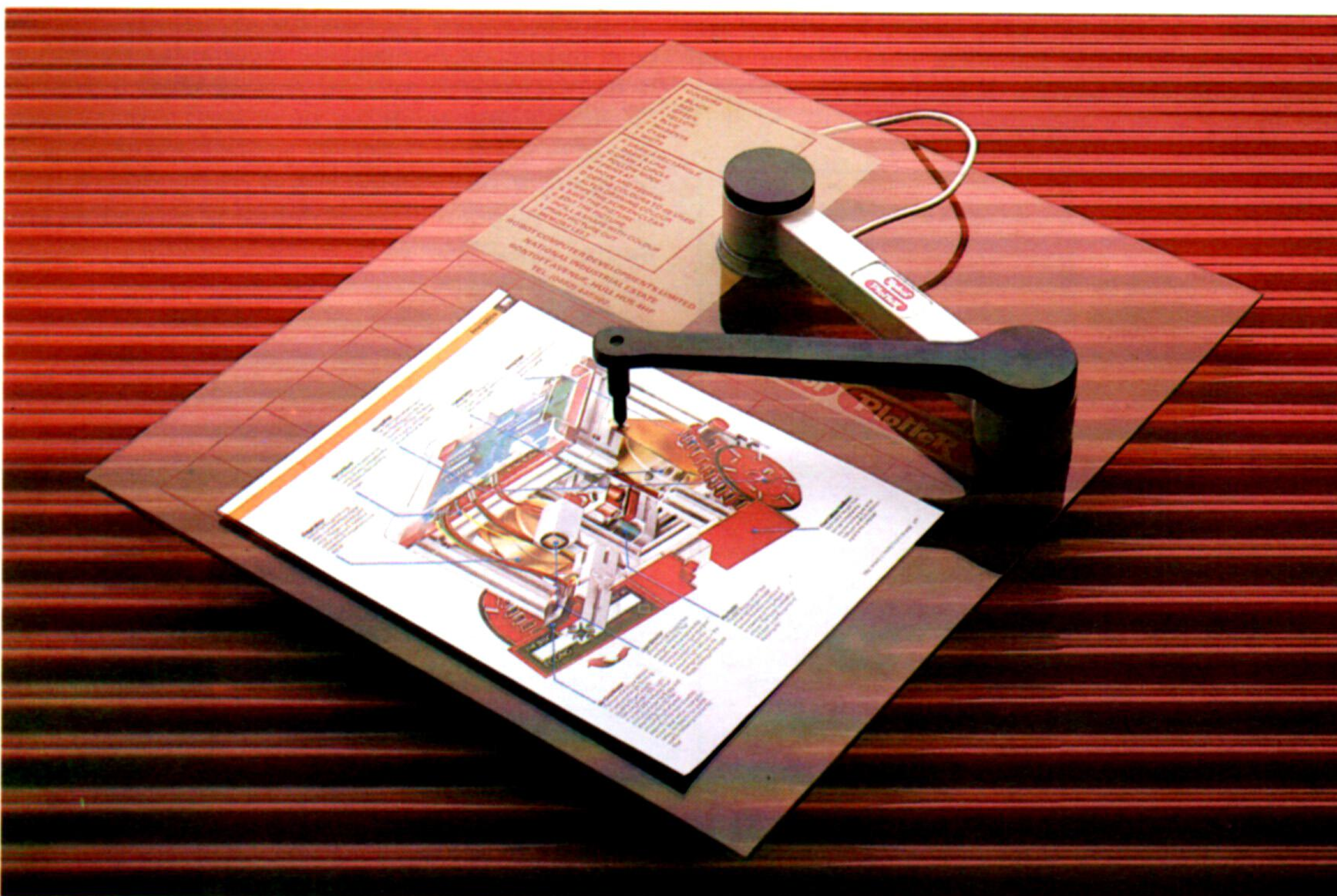
Todos los trazadores funcionan de forma similar. En el extremo de un brazo de doble articulación se fija un puntero que le envía señales eléctricas al ordenador. La intensidad de estas señales varía a tenor de la posición del puntero. El ordenador las convierte en un formato digital y las utiliza para trazar un punto en el lugar correspondiente de la pantalla. Todos los trazadores se suministran con software para llevar a cabo su tarea, ofreciendo diversas opciones, tales como dibujar líneas en distintos colores. El software para los modelos BBC permite seleccionar distintas modalidades de visualización, equilibrando la resolución y el número de colores disponibles dentro de las restricciones impuestas por el limitado espacio de memoria de que se dispone.

El Robot Plotter (trazador robot), de Robot Computer Development, es, de los cuatro que hemos examinado, el de aspecto más impresionante. Este modelo posee una base de perspex que lleva grabado un cuadriculado que muestra las posiciones de los pixels. El brazo trazador está sujeto en uno de los extremos de la base. La imagen a calcar se puede colocar debajo de la cuadrícula y contemplar a través de la base translúcida. El brazo de este modelo es sumamente sólido, construido con metal resistente y plástico. El puntero es una pluma parecida a un lápiz que une el brazo con la placa de la base. Lamentablemente, con este sistema no es fácil ver la imagen a medida que se la va calcando.

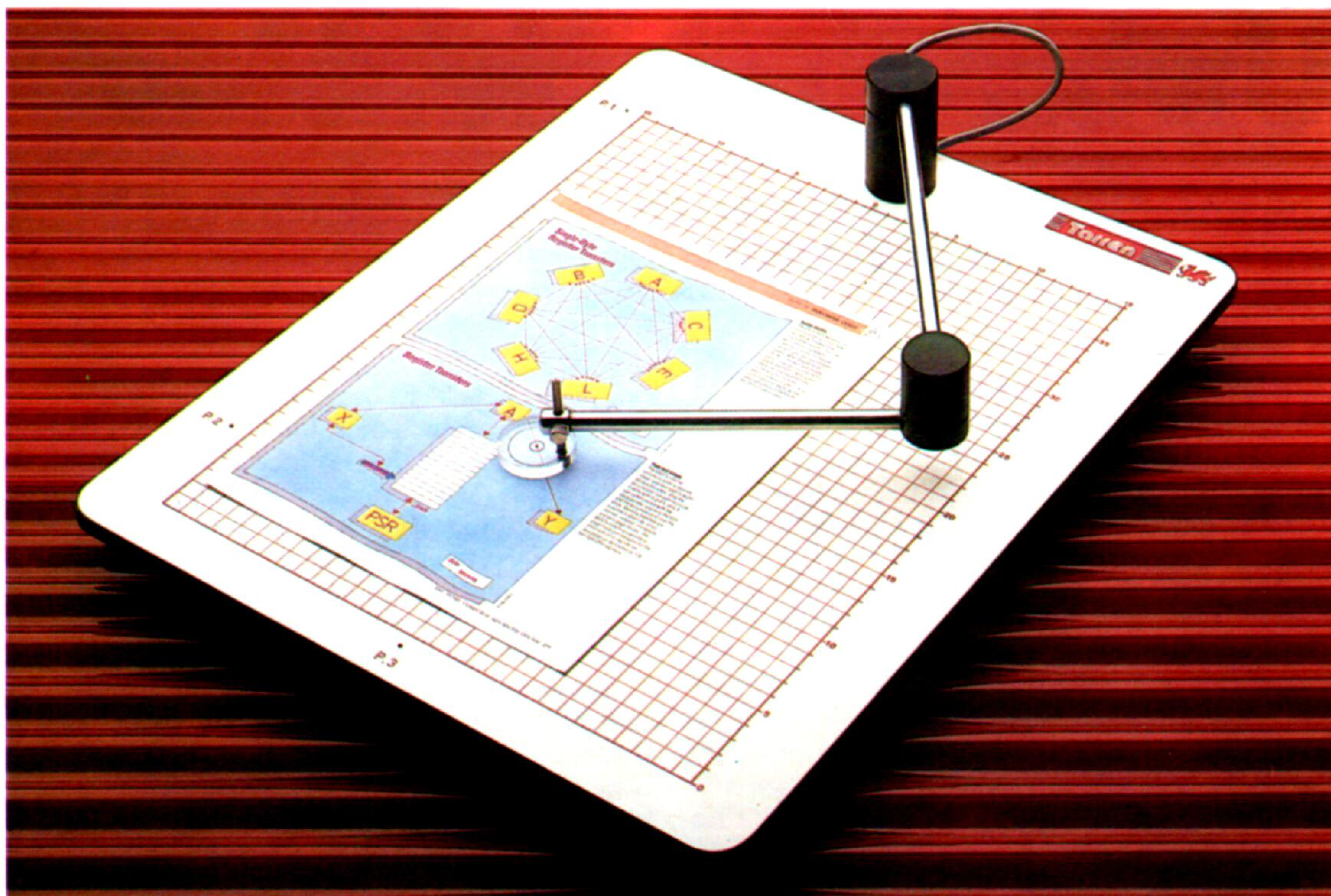
El Robot Plotter se vende con una cassette que contiene software para ejecutar en el BBC Micro. Además de las rutinas para calcar, la cassette contiene varias rutinas para dibujar círculos, rectángulos y líneas que se utilizan junto con el brazo.

El programa de trazado almacena todas las imágenes como una serie de líneas; por lo tanto, el trazado de un mapa se almacenará en la memoria como una secuencia de líneas cortas. Esto hace que resulte sumamente fácil eliminar líneas no deseadas sin afectar a las líneas cercanas. Sin embargo, una imagen compleja requiere muchísima memoria y es posible agotar todo el limitado espacio de memoria de la máquina BBC. Debido a que la visualización se almacena como una secuencia de líneas, es relativamente sencillo transferir imágenes creadas con el trazador a otro programa.

El trazador Digigraph también es de construcción sólida. La placa que sirve de base se compone de un gran tablero de madera en cuya superficie hay pintado un cuadriculado rojo. El brazo es de



Robot Plotter
Los originales a calcar se pueden colocar debajo del tablero de trabajo de perspex del Robot Plotter; el software que se suministra incluye rutinas para círculos y rectángulos



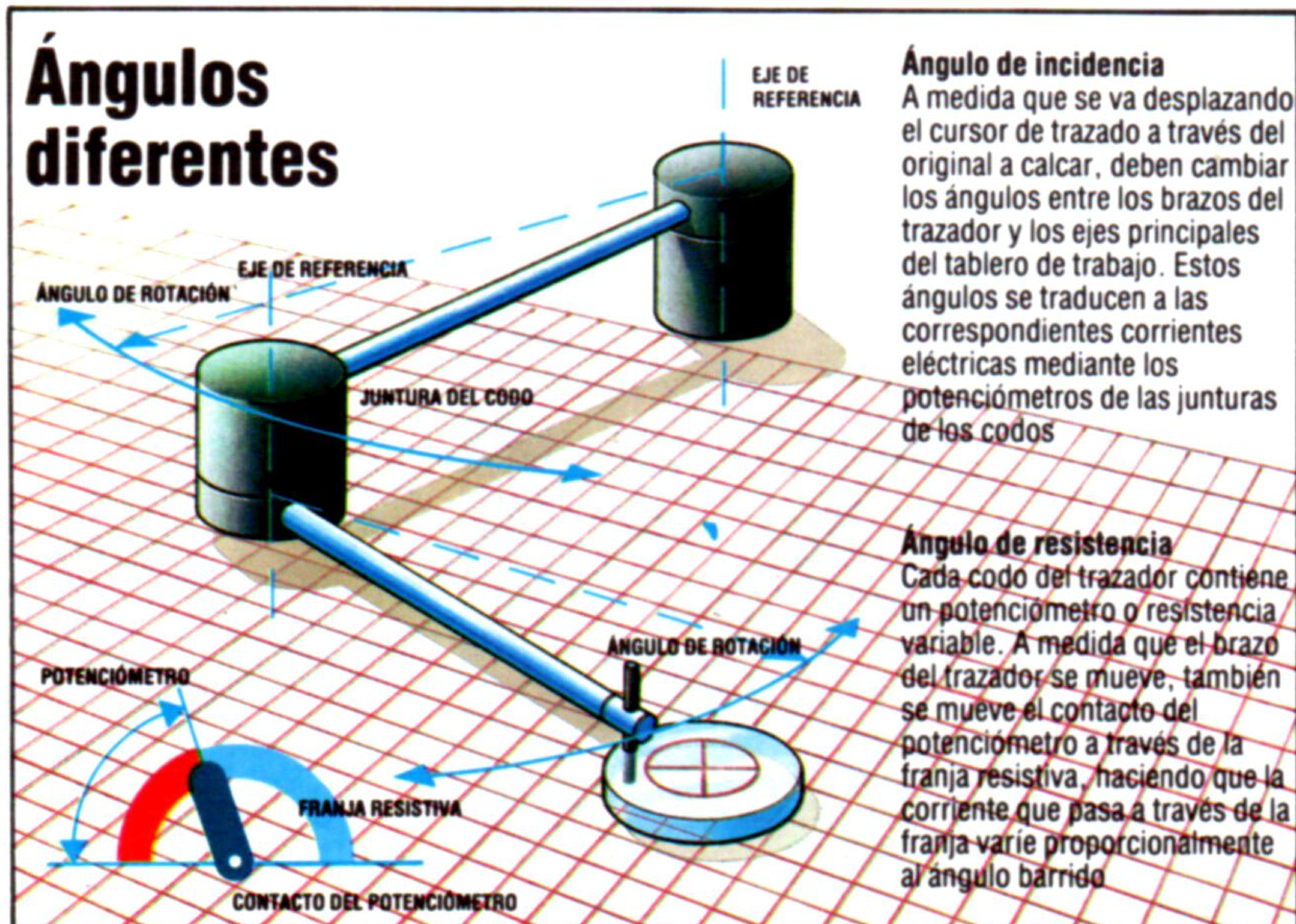
Digigraph
Sólidamente construido en metal y plástico, con un tablero para calcar de madera, el paquete Digigraph incluye varias hojas de trabajos para realizar prácticas

tubos de aluminio, con un disco de perspex en el extremo del puntero. La imagen a calcar se coloca sobre el tablero. Este sistema es el más fácil de utilizar de los cuatro que analizamos en este capítulo; el brazo se mueve suavemente y, a través del disco de perspex, la imagen resulta claramente visible. El software suministrado es menos sofisticado que el que se vende con el Robot Plotter, pero posee facilidades similares. El ordenador no almacena los movimientos del trazador como líneas separadas, sino que las imágenes se dibujan directamente en la pantalla, y el medio para salvar y cargar imágenes no es otro que salvar y guardar la zona de memoria de pantalla. Esto significa que no se puede editar fácilmente una imagen, pero posee la ventaja de que no se utiliza más memoria para dibujar una imagen compleja que para dibujar una imagen sencilla, lo que es muy útil tratándose del BBC Micro, tan escaso de memoria. El sistema Digigraph también incluye varias "hojas de trabajo", que el usuario puede usar para hacer prácticas con el trazador. Junto con los excelentes manuales, estas hojas permiten dominar el sistema dedicando a tal efecto un mínimo esfuerzo.

El trazador RD Labs se comercializa en dos formas: una para el BBC y otra para el Spectrum. Ambas versiones son, virtualmente, kits para ser montados en casa: sólo se suministra el brazo tiralíneas, junto con un material de relleno autoadhesivo que permite pegar el brazo a una base apropiada. El brazo es de material plástico y muy flexible, pero funciona con razonable precisión. El puntero es un dispositivo plástico con dos rayas cruzadas cuya utilización es más bien incómoda.

El precio de la versión para el BBC incluye el brazo propiamente dicho, un manual y una cassette con software. Las imágenes se almacenan de forma similar al Robot Plotter, con el resultado de que se vuelve a plantear el problema de la escasez de memoria. El software es bastante complejo y contiene rutinas para dibujar círculos, rectángulos y líneas así como una facilidad para animación. Ésta utiliza la capacidad del BBC de redefinir la paleta de colores para animar secuencias breves y simples. Una cassette de demostración descubre todas las características de este trazador.

La versión para el Spectrum es la más barata de las cuatro. El paquete incluye un accesorio convertidor de analógico a digital (el BBC Micro posee uno ya incorporado). En la mayoría de los otros sentidos, el modelo para el Spectrum es muy similar a la versión para el BBC. Se puede obtener un dibujo continuo, pero es difícil hacerlo satisfactoriamente debido a la baja respuesta del software, y una curva suave tiende a visualizarse en la pantalla como una serie de líneas rectas. No obstante, están disponibles el resto de facilidades (para dibujar círculos, rectángulos, etc.). El software no produce un aviso de error (a diferencia de los programas nor-



Ángulos diferentes

Ángulo de incidencia
A medida que se va desplazando el cursor de trazado a través del original a calcar, deben cambiar los ángulos entre los brazos del trazador y los ejes principales del tablero de trabajo. Estos ángulos se traducen a las correspondientes corrientes eléctricas mediante los potenciómetros de las junturas de los codos

Ángulo de resistencia
Cada codo del trazador contiene un potenciómetro o resistencia variable. A medida que el brazo del trazador se mueve, también se mueve el contacto del potenciómetro a través de la franja resistiva, haciendo que la corriente que pasa a través de la franja varíe proporcionalmente al ángulo barrido





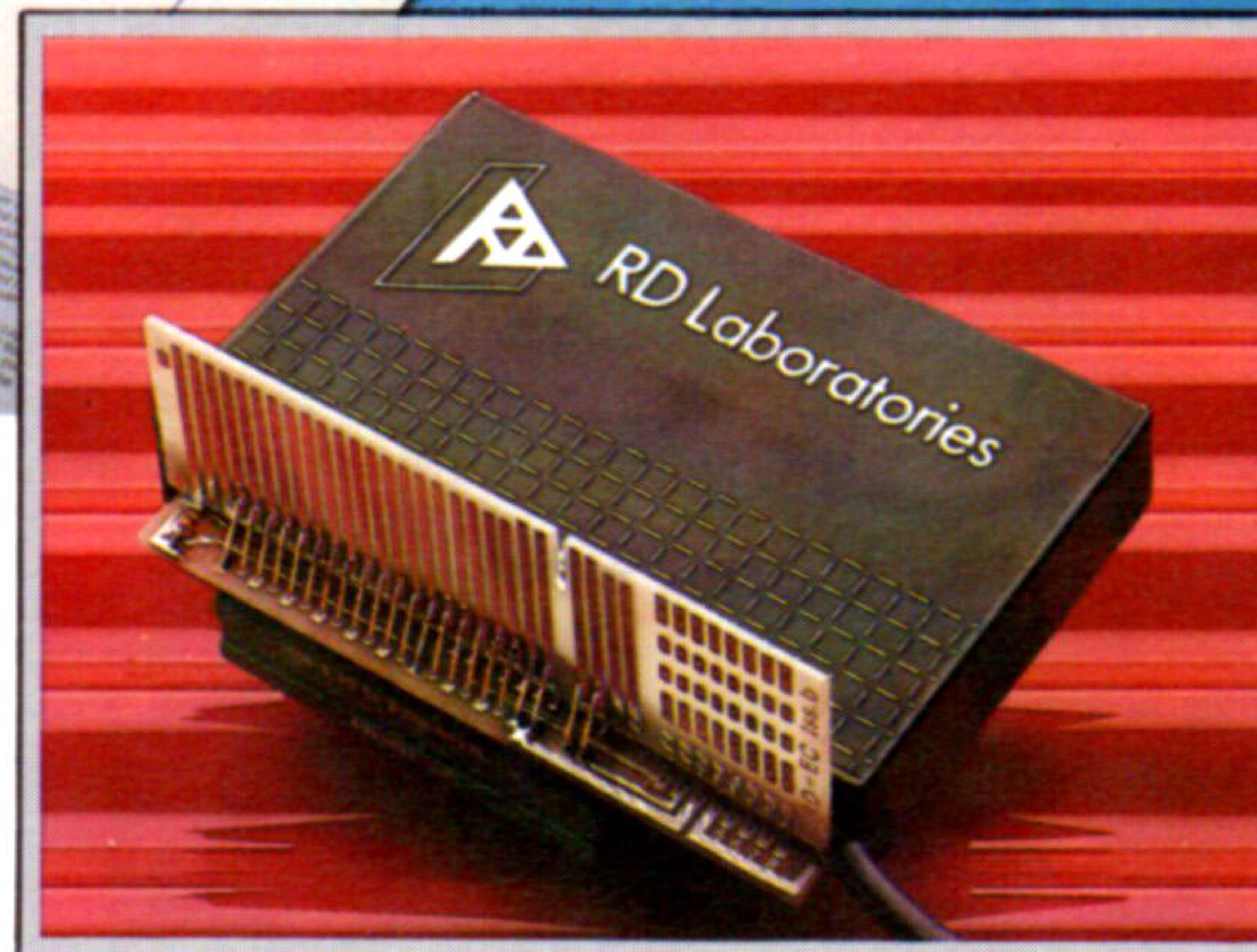
Trazador digital RD
Esta versión para el BBC Micro está construida con sencillez en plástico ligero; se proporciona una banda de cinta adhesiva, que no se puede quitar una vez usada, para fijarlo a la mesa de trabajo

males para el Spectrum) si se utilizan estas instrucciones para trazar líneas que salen de los límites de la pantalla.

Es posible escribir programas que lean posiciones directamente de cualquiera de estos trazadores, lo que resulta útil para ciertos fines especiales. No obstante, esto es difícil de conseguir en el Spectrum, porque este ordenador, al contrario del BBC, no posee puerta para palanca de mando y su BASIC carece de las instrucciones pertinentes.

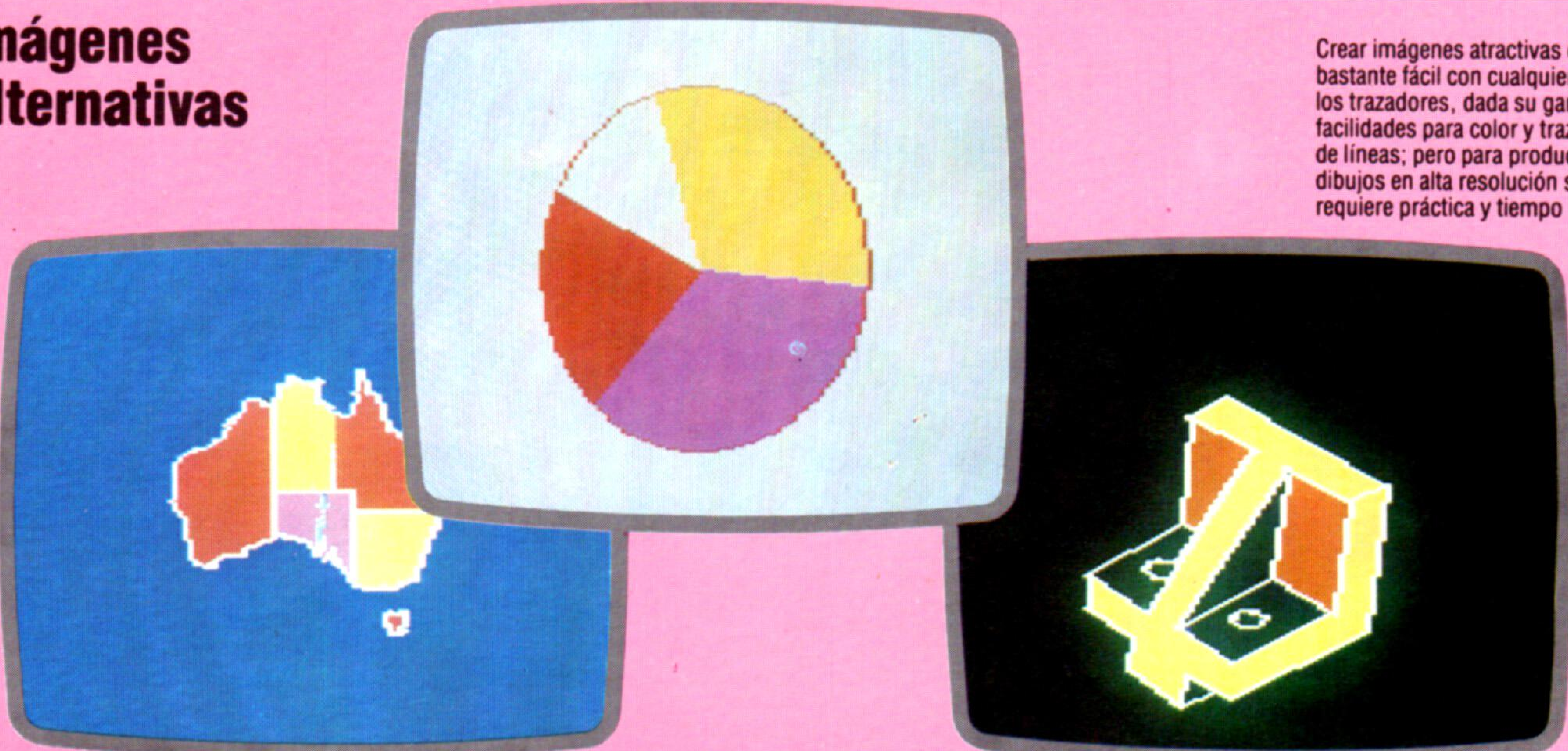
Para los usuarios de un Spectrum, la elección de un trazador se limita al modelo RD Labs, pero éste trabaja suficientemente bien. El usuario del BBC dispone de una gama de opciones más amplia: el tiralíneas RD Labs es el más económico de los tres, pero su construcción es, en cierta manera, endeble; el Robot Plotter está construido sólidamente y tiene el apoyo de un software inteligente, si bien no es particularmente fácil de utilizar; el Digigraph tiene un software menos sofisticado pero es, con mucho, el más fácil de utilizar.

Interface RD para el Spectrum



El trazador RD para el Spectrum es exactamente igual a la versión para el BBC cuya fotografía se incluye en este capítulo, pero se conecta mediante su propia interface

Imágenes alternativas



Crear imágenes atractivas es bastante fácil con cualquiera de los trazadores, dada su gama de facilidades para color y trazado de líneas; pero para producir dibujos en alta resolución se requiere práctica y tiempo

Consulta privada

La comunicación con los ordenadores mediante el lenguaje corriente será pronto una realidad. Veamos un programa que ilustra sus principios

Uno de los pioneros de la informática, Alan Turing, propuso un famoso test de inteligencia artificial: afirmó que si una máquina es capaz de conversar con un ser humano sin que éste pueda darse cuenta de que está hablando con una máquina, entonces se puede afirmar que esa máquina es "inteligente". Hasta ahora, no hay ninguna máquina que haya superado esta prueba con total éxito, pero existen varios programas que se acercan a la misma apartando el interés del hombre por la conversación de las respuestas de la máquina, de manera que sea éste quien mantenga la conversación. Esto puede parecer rebuscado, pero resulta sorprendentemente eficaz. Intente recordar alguna conversación que haya mantenido y le haya producido satisfacción, y probablemente descubrirá que fue usted quien habló la mayor parte del tiempo, mientras

Programa

```
100 GOSUB 2000: REM INIC
200 FOR L = 1 TO 2*LT STEP 2
300 PRINT O$: INPUT I$
400 GOSUB 3000: REM ANALISIS
500 NEXT L
1000 PRINT TAB(5);"-----LA SESION HA TERMINADO-----"
1100 PRINT TAB(3);"-----PULSE CUALQUIER TECLA PARA
     CONTINUAR-----"
1200 A$ = INKEY$: IF A$ = "" THEN GOTO 1200
1300 GOSUB 5000: REM INFORME
1900 END
1999 REM ***************************************
2000 REM **            S/R INICIALIZAR        **
2001 REM ***************************************
2050 LT = 10: AN = 10: T9 = 500: EX = 2*LT
2100 DIM R$(AN): DIM H$(2*LT)
2200 DATA "SI..","DIGAME ALGO
     MAS..","CONTINUE..","Y..","POR TANTO.."
2250 DATA "ES ESO IMPORTANTE..","POR QUE LE
     PREOCUPA ESO.."
2300 DATA "POR FAVOR EXPLIQUEME ESO..","POR QUE LO
     DICE.."
2350 DATA "Y ESO EN QUE LO AFECTA A USTED.."
2500 FOR K = 1 TO AN: READ R$(K): NEXT K
2600 CLS: O$ = "HOLA - CUAL ES SU PROBLEMA.."
2950 RETURN
2999 REM ***************************************
3000 REM **            S/R ANALISIS           **
3001 REM ***************************************
3100 IF I$ = "ADIOS" THEN EX = L-1: L = 2*LT: RETURN
3200 H$(L) = O$: H$(L + 1) = I$
3300 R9 = INT(AN*RND + 1): IF R9 = R0 THEN GOTO 3300
3400 O$ = R$(R9): R0 = R9
3950 RETURN
4999 REM ***************************************
5000 REM **            S/R INFORME            **
5001 REM ***************************************
5050 CLS: PRINT TAB(10);"**INFORME**"
5100 FOR K = 1 TO EX STEP 2
5150 PRINT TAB(5);H$(K)
5200 PRINT H$(K + 1)
5300 FOR D = 1 TO T9: NEXT D
5400 NEXT K
5900 RETURN
```



Michael Brownlow



que la otra persona tan sólo se limitó a proporcionar "ruidos" o señales conversacionales. Éste es el principio sobre el que se basan muchas clases de psicoterapias y asesoramientos.

El programa que listamos aquí simula el comportamiento de un terapeuta completamente no directivo o, dicho de otra manera, un terapeuta que no *dirige* la conversación, sino que mantiene el flujo de la misma mediante respuestas tales como "HÁBLEME MÁS..." y "¿POR QUÉ ESO ES IMPORTANTE?" Conserva una grabación de la charla y la reproduce después de cierto número de intercambios, o bien cuando usted digita "ADIÓS".

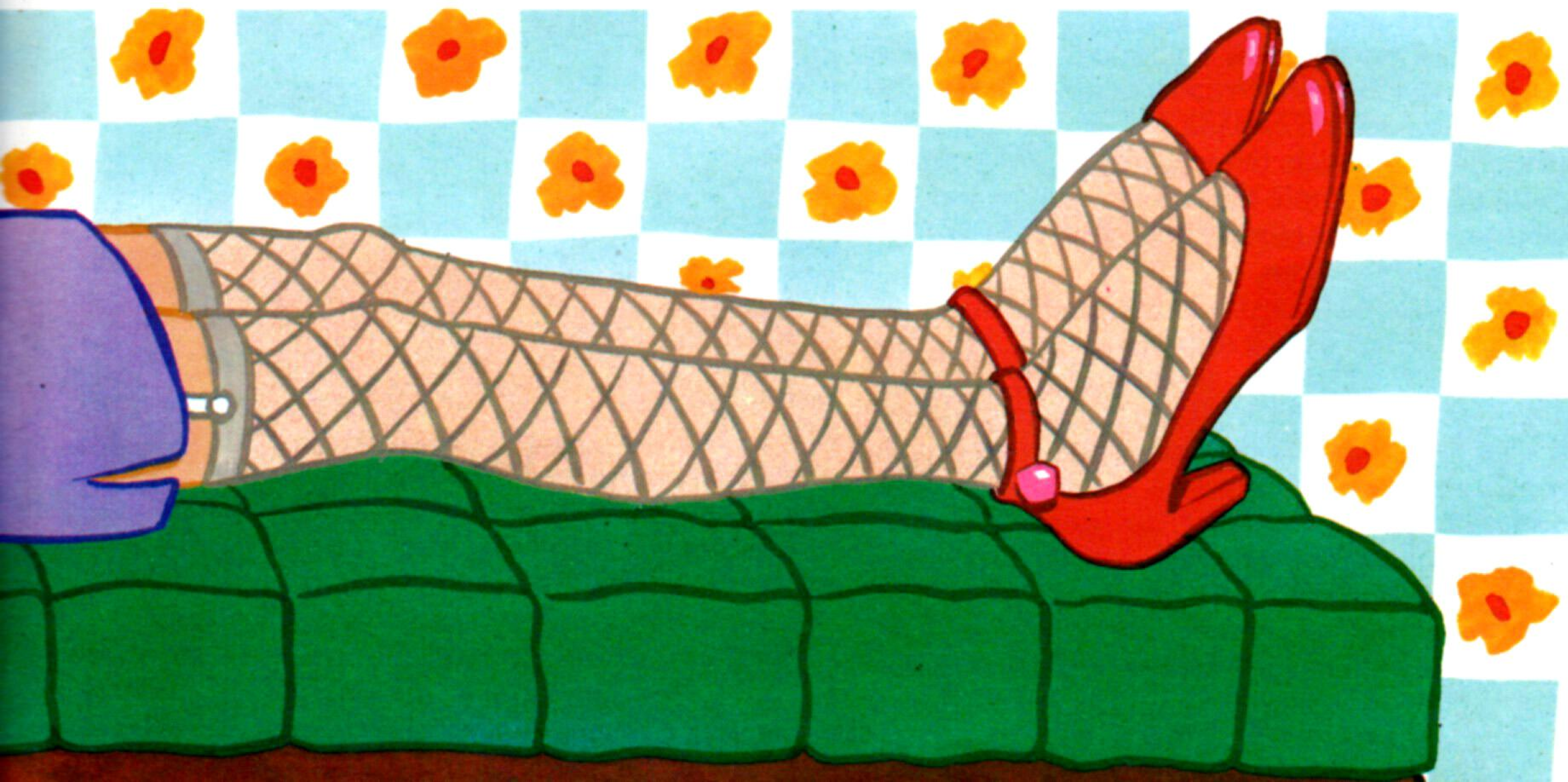
El programa se puede desarrollar en la dirección de la pseudointeligencia haciendo que analice las entradas del usuario y elija una respuesta adecuada. Esto ya lo hace en la línea 3100 verificando si la palabra es "ADIÓS". Nosotros podríamos ampliar este análisis verificando las palabras "SÍ" y "NO" y produciendo señales más específicas en las respuestas; algo tan sencillo como "¿POR QUÉ?" o "¿POR QUÉ NO?" funcionaría muy bien. A continuación podríamos comprobar si el usuario está repitiendo una respuesta y, de ser así, replicarle en consecuencia o dar por finalizada la sesión. Podríamos verificar si una respuesta termina en un signo de interrogación o de exclamación, y replicar "¿POR QUÉ ME PREGUNTA ESO?" o "¿POR QUÉ SE EXCITA?"

Ahora podríamos preparar una tabla de palabras clave y buscar la respuesta para estas palabras, produciendo una respuesta específica tomada de otra tabla en caso de encontrarla. La selección de palabras clave y de respuestas depende del tipo de conversación que espere mantener; el modo en que su selección afecte a las respuestas del usuario le puede proporcionar algunos indicios esclarecedores acerca de su propio subconsciente o del subconsciente de sus amigos. El inconveniente de este método (y el de todos los métodos que deben buscar el texto) es el tiempo que lleva analizar incluso la oración más corta. En este caso el factor limitante es la velocidad del BASIC, y todo análisis serio exige un programa en lenguaje máquina. Podemos, sin embargo, tolerar demoras en aras de la investigación.

Si pretendemos analizar las palabras del usuario, entonces son muchos los enfoques que podemos adoptar; probablemente el más interesante de éstos sea el *análisis sintáctico*, o sea la reducción de una oración en las partes que la componen, como pronombre, verbo o sustantivo. Ello exige un cuerpo de reglas sintácticas y gramaticales, y tablas de pronombres, preposiciones, conjunciones, transformaciones de palabras, etc., y no es una cuestión sencilla. No obstante, sería fácil elegir la palabra más larga de la respuesta y pedirle al usuario que explique las sensaciones que la misma le produce; es probable que la palabra más larga, en el caso de una oración simple, sea también la más importante. Tal vez pudiéramos introducir una mejora eligiendo al azar una palabra entre todas las palabras que tengan más de, pongamos por caso, cinco letras, o bien utilizar la palabra que aparezca a continuación de "YO", o "MI" o "SU".

Elegir uno de estos métodos y refinarlo es un fascinante ejercicio de programación. Se obtiene un panorama muy claro sobre la enorme complejidad del lenguaje hablado y de su análisis, y puede llegar a sentirse insatisfecho con el BASIC como herramienta analítica. Ello le puede conducir a acercarse a otros lenguajes de programación, tales como el LISP y el PROLOG, que están estructurados de forma muchísimo más sutil, precisamente debido a la necesidad de hacer frente a la complejidad del lenguaje hablado y el pensamiento. Además de esto, por supuesto, también tendrá la oportunidad de hablar todo el tiempo que lo desee con el conversador perfecto: ¡alguien que siempre lo escucha!

```
3420 IF LEFT$(I$,2) = "SI" THEN O$ = "POR QUE ESTA UD
     TAN SEGURO..": RETURN
3450 IF LEFT$(I$,2) = "NO" THEN O$ = "POR QUE NO..":
     RETURN
3500 Z$ = RIGHT$(I$,1)
3520 IF Z$ = "?" THEN O$ = "POR QUE ME LO
     PREGUNTA.."
3550 IF Z$ = "!" THEN O$ = "POR QUE LO PERTURBA
     ESO.."
3600 IF R9 < AN/4 THEN GOSUB 4000
3950 RETURN
3999 REM ****************************************
4000 REM **          S/R PALABRA MAS LARGA        **
4001 REM ****************************************
4050 W = 1: H = 1: WL = 1: S = 1
4100 I$ = I$ + " ": L9 = LEN(I$)
4120 FOR C = 1 TO L9: FOR P = C TO L9
4150 Z$ = MID$(I$,P,1)
4170 IF Z$ = " " THEN W = P-C: H = C: C = P: P = L9
4200 NEXT P
4200 IF W > WL THEN WL = W: S = H
4250 NEXT C
4270 O$ = "QUE SIGNIFICA " + MID$(I$,S,WL) + "PARA
     UD.."
4450 RETURN
```



```
      **INFORME**
      HOLA—CUAL ES SU PROBLEMA
EL HALCON NO PUEDE OIR AL HALCONERO
      DIGAME ALGO MAS..
LAS COSAS SE SEPARAN — EL CENTRO NO AGUANTA
      CONTINUE..
LA MERA ANARQUIA SE LIBERA POR EL MUNDO
      POR FAVOR EXPLIQUEME ESO..
LOS MEJORES CARECEN DE CONVICCIONES
      ES ESO IMPORTANTE..
LOS PEORES ESTAN LLENOS DE APASIONADA
   INTENSIDAD SI..
SEGURAMENTE HAY UNA REVELACION A MANO
      POR FAVOR EXPLIQUEME ESO..
SEGURAMENTE LA SEGUNDA VENIDA ESTA AL
   LLEGAR PORQUE ES ESO IMPORTANTE...
Y QUE TORPE BESTIA — SU HORA LLEGO AL FIN
      CONTINUE..
CAMINA HACIA BELEN PARA NACER
```

Pensamientos estimulantes
Hemos utilizado algunos versos del poema de W. B. Yeats *The second coming* (La segunda venida) para ilustrar la capacidad de un programa para estimular al usuario a pensar

Agregue estas líneas al primer programa para detección de SI/NO y de la palabra más larga

Complementos al BASIC
Este programa está escrito en BASIC Microsoft. Los usuarios del Spectrum deben insertar LET en todas las sentencias de asignación y ajustar los valores de TAB

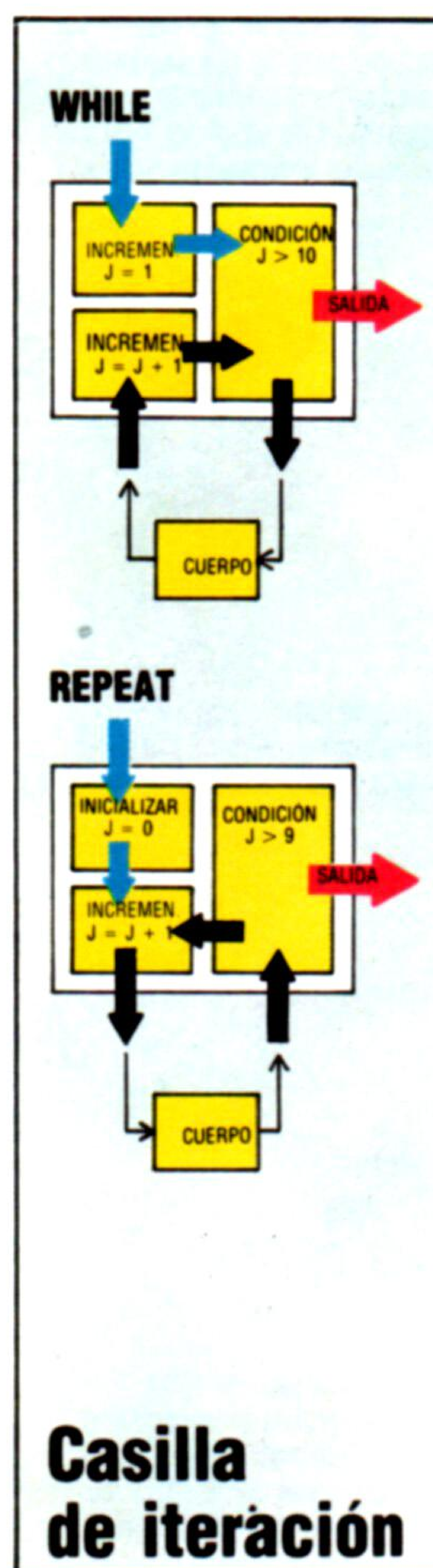
Spectrum
Reemplazar DIM R$(AN):DIM H$(2*LT) por DIM R$(AN,30):DIM H$(2*LT,100)
Reemplazar LEFT$(I$,2) por I$(TO 2)
Reemplazar RIGHT$(I$,1) por I$(LEN(I$))
Reemplazar MID$(I$,P,1) por I$(P)
Reemplazar MID$(I$,S,WL) por I$(S TO S + WL−1)

Commodore Vic-20 y 64
Reemplazar CLS por PRINT CHR$(147)
Reemplazar INT(AN*RND + 1) por INT(RND(1)*AN + 1)
Reemplazar A$ = INKEY$ por GET A$

BBC Micro
Reemplazar A$ = INKEY$ por A$ = INKEY$(0)
Reemplazar INT(AN*RND + 1) por RND (AN)

Líneas de bucle

En esta ocasión analizaremos diversas formas de clasificar los bucles e introduciremos un nuevo símbolo para diagramas de flujo: la casilla de iteración



Casilla de iteración

La iteración, o bucle, es una de las estructuras esenciales de todos los lenguajes de programación. En el curso ya hemos dicho anteriormente que en un algoritmo se utiliza un bucle cada vez que una decisión dirige el flujo de control por un camino que, a la larga, lo lleva de regreso a la misma decisión inicial. Esto describe adecuadamente la estructura: la ejecución repetida de un cuerpo de código. Sin embargo, esta definición no abarca todas sus formas, que son muchas. Dado que los bucles son una estructura primitiva tan esencial que probablemente comprendan el 60 % de toda la actividad del procesador, es sumamente útil analizarlos en mayor detalle. Prestaremos especial atención al efecto que ejercen sobre la estructura del programa-algoritmo general y a las diversas formas en las que se pueden construir y clasificar.

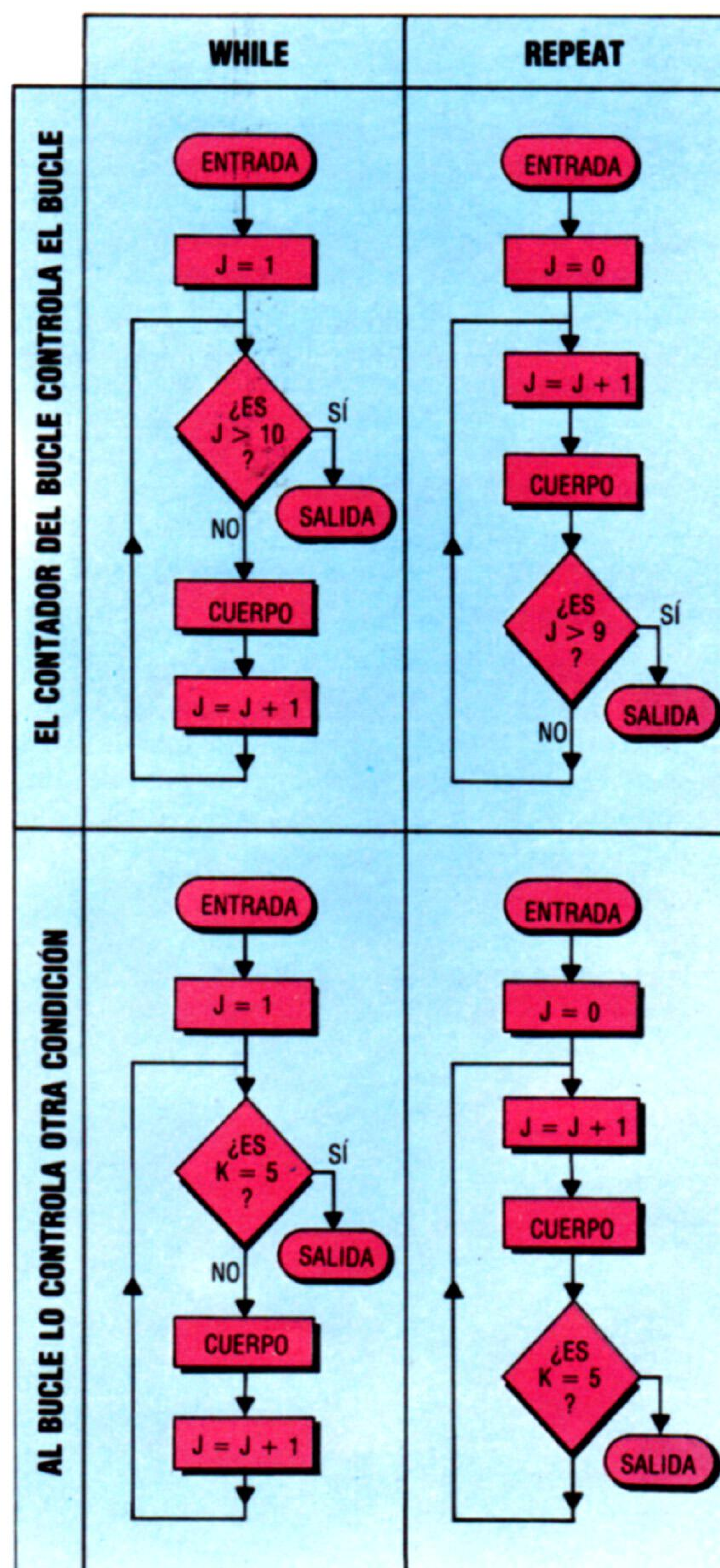
Los bucles se suelen dividir en dos clases, según su similitud con las dos estructuras de bucles de los lenguajes de alto nivel, RETURN...UNTIL y WHILE...ENDWHILE; ambos tipos se utilizan en PASCAL, y el bucle REPEAT está implementado en el BASIC del BBC y del Oric. Los dos tipos difieren en la colocación de la condición de salida del bucle: en un bucle REPEAT la condición va al final del cuerpo del bucle, mientras que en un bucle WHILE va al principio del mismo. Esto significa que el cuerpo de un bucle REPEAT, una vez se ha entrado en él, siempre se ejecutará al menos una vez, mientras que el cuerpo de un bucle WHILE no se ejecutará necesariamente. Esta primera diferencia a la que nos referimos se puede apreciar con bastante claridad en el diagrama de flujo.

Otra forma de clasificar los bucles atiende al hecho de si la variable que actúa como contador del bucle se utiliza en la condición de salida del bucle, o si hay alguna otra condición que controle esta salida. Esto es más difícil de ver en un diagrama de flujo convencional de tipo lineal. En realidad, la clase de diagrama de flujo de "sentido común" con el cual nos hemos familiarizado describe los bucles de una forma que no es nada útil. En estos diagramas de flujo, un bucle tiene exactamente el mismo aspecto que una simple bifurcación, y a menudo es necesario examinar el algoritmo con todo detalle para distinguir a ambos.

Existe una notación más clara llamada "casilla de iteración" que señala con toda claridad el comienzo de un bucle y elimina la confusión entre bucle y bifurcación. Consta de tres casillas unidas: la primera casilla indica la inicialización del contador, la segunda muestra el incremento del contador, y la tercera contiene la condición de salida del bucle. De esta forma se pueden reflejar tanto los bucles REPEAT como los WHILE. Ambos difieren en el flujo de control a través de las casillas, y la casilla de la condición indica si el bucle está controlado mediante el contador o no. Estos puntos se pueden apreciar claramente en el diagrama.

En un bucle REPEAT el flujo de control es "inicializar-cuerpo-condición-cuerpo-condición", mientras que el de un bucle WHILE es "inicializar-condición-cuerpo-condición-cuerpo". Esto se puede observar en la forma en que las líneas de control salen y vuelven a entrar en la casilla de iteración, con el cuerpo del bucle "colgando" de ellas.

Clasificación de los bucles





Billar informatizado

A primera vista el billar puede parecer un juego nada apropiado para la simulación por ordenador, pero la realidad es diferente...

A cualquiera que haya jugado alguna vez al *snooker* (billar) la idea de informatizarlo le parecerá un disparate. Para un jugador profesional constituirá un sacrilegio. ¿Cómo, se preguntarán, se podrían cuantificar y programar todas las sutiles curvas de la trayectoria de una bola o todas las variaciones de las características de la mesa? ¡No, por cierto, dentro de los límites de un Spectrum de 16 Kbytes! Sin embargo, se puede pensar en el juego del *snooker* como una aplicación relativamente sencilla del principio de la conservación del impulso, y un ordenador puede manipular las matemáticas del mismo con toda facilidad.

Cuando se comienza a jugar con el *Snooker* de Visions se visualiza en pantalla un plano de la mesa, con las bolas representadas por círculos de distintos colores. El jugador dirige la bola blanca desplazando una cruz por la pantalla hasta una posición determinada, definiendo de esta manera el camino que seguirá la bola. La bola blanca es disparada entonces hacia su objetivo con una fuerza que está determinada por la duración del período de tiempo durante el cual se mantiene pulsada la tecla.

Después de haber dado unos pocos golpes, un jugador avezado se dará cuenta de dos importantes problemas. El primero de ellos es que el ordenador (ya sea un Spectrum, un BBC o un Commodore) no puede calcular los ángulos de movimiento ni la resistencia por rozamiento suficientemente bien como para dar resultados convincentes en una pantalla de televisión. Ello significa que muchos golpes dan resultados bastante imprevisibles (si bien esto también puede suceder en el juego real). El jugador se dará cuenta de que no puede dar ciertos golpes, que son imposibles en el ordenador. Para compensar esto, las troneras son proporcionalmente más grandes de lo que deberían ser.

El otro problema, que en la práctica es más profundo, es que el jugador carece de perspectiva en su visión de la mesa. Ello significa que cuando le toca dar un golpe se encuentra privado de la ventaja que significa poder mirar a lo largo del taco. Y debido a que una pantalla de televisión es ligeramente curva, juzgar los ángulos resulta el doble de difícil. Es probable que, por todo cuanto acabamos de exponer, los primeros intentos que realice con este juego sean sumamente frustrantes.

No obstante, después de haberse aceptado el hecho de que el juego se parece muy poco al billar auténtico y una vez se han dominado algunas de sus pocas excentricidades, *Snooker* se convierte por derecho propio en un juego de estrategia fascinante. Se comienzan a apreciar algunas sutiles características de la programación, en especial la posibilidad de darle efecto a la bola, aunque los resultados en este caso vuelven a ser impredecibles.

Las tres versiones del juego se diferencian en formas que reflejan los puntos fuertes y los puntos débiles de las máquinas con las que se está jugando. Sorprendentemente, el juego utiliza poca memoria y, por lo tanto, se carga rápidamente en las tres máquinas, incluso en el Commodore, tan conocido por su baja velocidad de carga. Así que la calidad de cada versión refleja las capacidades para gráficos de cada máquina y no su capacidad de memoria.

En este sentido el Spectrum es el más flojo. La mesa es pequeña, y no todas las bolas se visualizan con sus colores auténticos. La versión para el Commodore es mejor, con una mesa más grande, colores realistas y bolas representadas a una escala más exacta. En esta versión el marcador está ilustrado de forma más llamativa y al comienzo del juego se ofrece una opción de demostración (aunque debe completar una partida antes de que se le devuelva el control). La versión para el BBC es la mejor de las tres, con una mesa que cubre virtualmente todo el ancho de la pantalla y fortalecida por un control del taco mucho más preciso.

El juego posee capacidad para uno o dos jugadores, si bien la opción de un único jugador sólo tiene sentido con fines de aprendizaje. Dado que existen tantos juegos por ordenador sumamente antisociales, este juego para dos jugadores es muy valioso en este sentido. Lamentablemente, Visions no ha creído conveniente proporcionar con el software los batines clásicos. El juego da la impresión general de ser digno de consideración, pero cabe la sospecha de que no se ha sabido aprovechar al máximo su potencial, en especial por haber tanta memoria sin utilizar.

Snooker: Para el Commodore 64, el Spectrum de 16 K y el BBC Micro
Editado por: Visions (Software Factory) Ltd, 1 Felgate Mews, Studland Street, London
Autor: Tim Bell (adaptación de Andy Williams para el BBC)
Palancas de mando: Opcionales
Formato: Cassette

La esencia del billar es apuntar con precisión y calibrar con exactitud los tiros. En el *snooker* de Visions se debe situar en la mesa un cursor de puntería mientras se controla la fuerza del golpe a base de pulsar la tecla de disparo más o menos tiempo

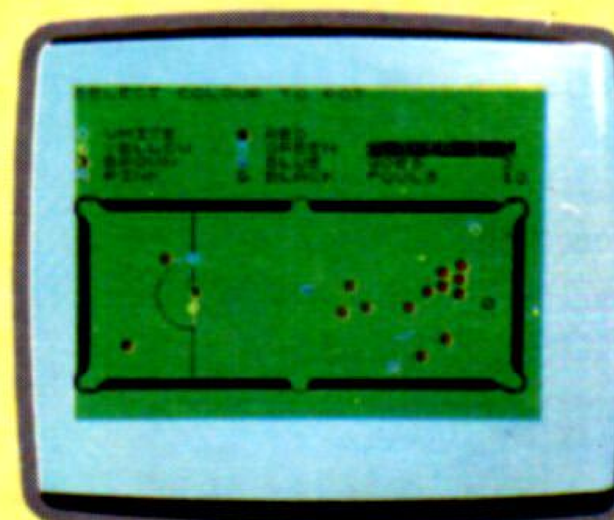
De esquina a esquina

BBC Micro



Commodore 64



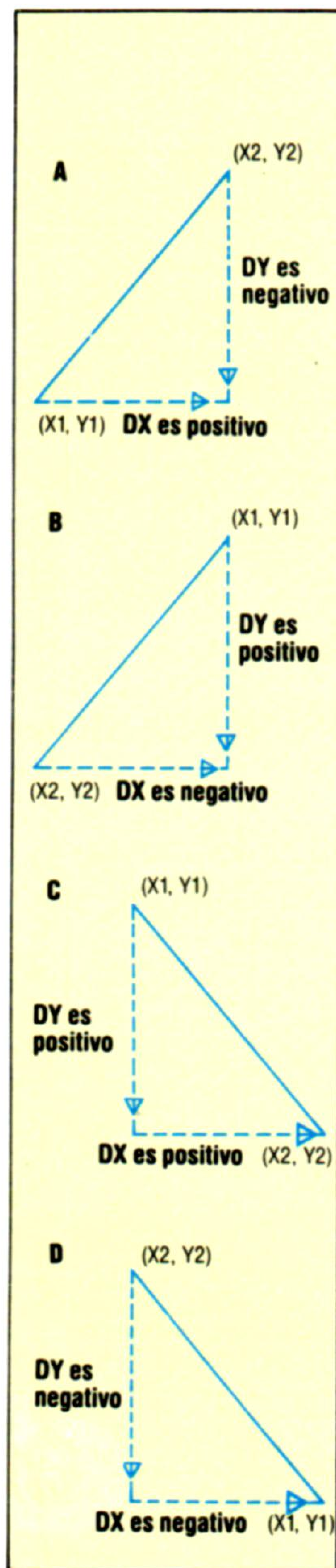


Spectrum

Ian McKinnell

Rectas y pendientes

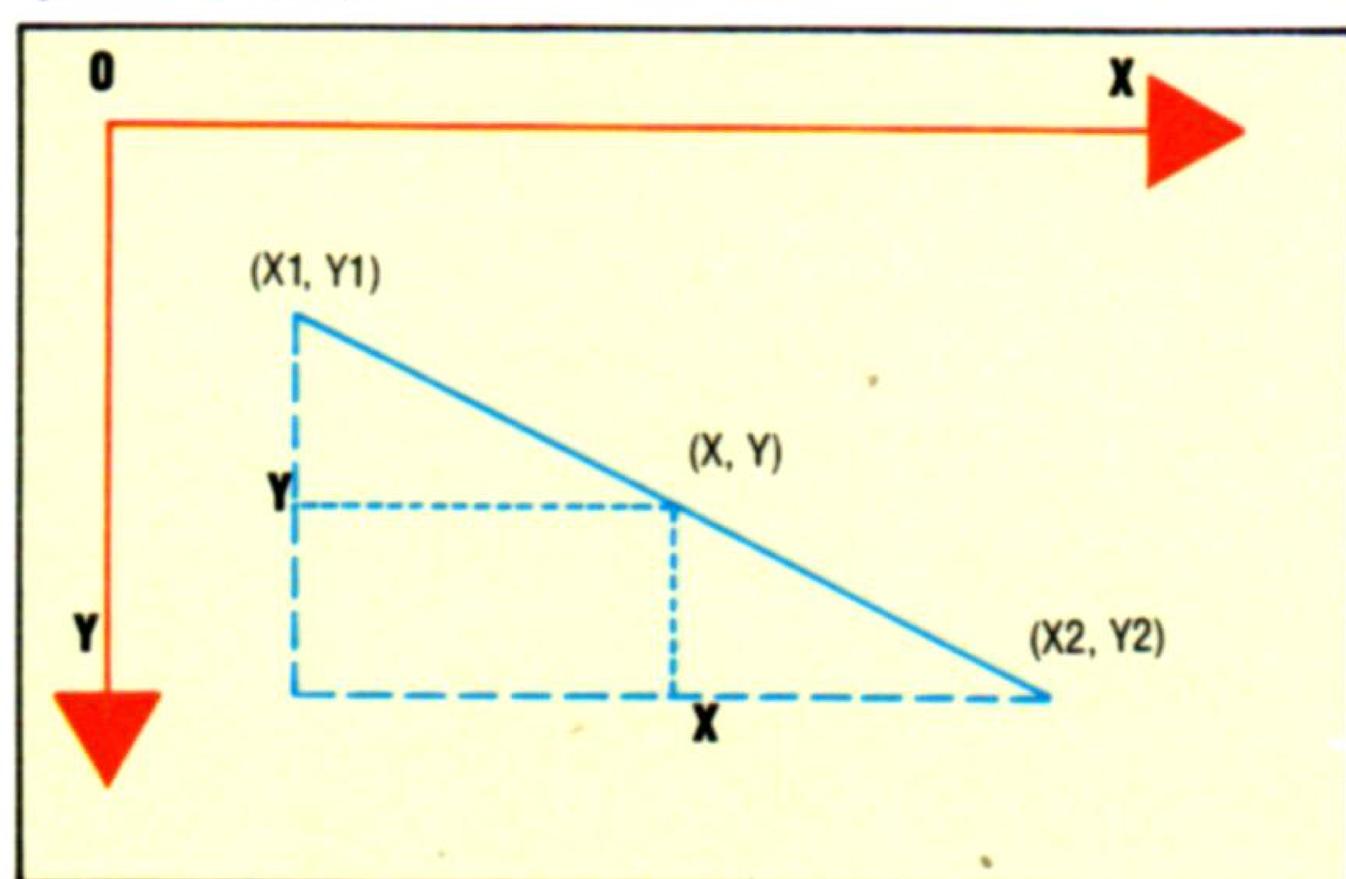
Ya ofrecimos una rutina para el Commodore 64 en lenguaje máquina. La que aquí estudiamos nos servirá para trazar todo tipo de líneas



Cuatro en uno
Cuando el valor absoluto (sin signo) de la pendiente (Y2−Y1)/(X2−X1) es mayor que uno conviene que DX sea positivo, y para ello emplearemos uno de estos algoritmos:
A) Volver a calcular —(DY y decrementar Y a partir de (X1,Y1) hasta alcanzar Y2
B) Intercambiar (X1,Y1) con (X2,Y2) y repetir cálculos
C) Incrementar Y partiendo de (X1,Y1) hasta alcanzar Y2
D) Intercambiar (X1,Y1) con (X2,Y2) y repetir los cálculos

La manera más obvia de trazar una línea de un punto a otro es calculando su pendiente según las coordenadas de los extremos. Así podemos pasar del punto de partida al siguiente mediante un incremento de la abscisa X y otro incremento proporcional de la ordenada Y, que obtendremos de acuerdo a la pendiente. De este punto pasaremos a trazar los sucesivos, hasta llegar al otro extremo de la línea. Esto supone unas sencillas nociones matemáticas que ahora exponemos.

Llamaremos a los puntos extremos (X1,Y1) y (X2,Y2) (no se olvide que un punto está definido por sus *dos* coordenadas), y pendiente P al cociente (Y2-Y1)/(X2-X1). El dibujo muestra una línea con sus puntos extremos y otro punto cualquiera (X,Y):



La pendiente de la semirrecta que va desde el punto inicial (X1,Y1) a ese punto cualquiera es P = (Y−Y1)/(X−X1) que, naturalmente, es idéntica a la calculada para la recta entera. Recordando cómo se despeja en las ecuaciones podemos obtener el valor de la ordenada Y:

$$(Y-Y1)/(X-X1) = P$$
$$(Y-Y1) = (P(X-X1)$$
$$Y = Y1 + P(X-X1)$$

En el caso de que el paso de X a X1 mida la unidad (o sea, los incrementos de la abscisa X valgan uno), entonces

$$Y = Y1 + P$$

En conclusión: si comenzamos por el extremo (X1,Y1) y vamos incrementando la X, los valores sucesivos de la Y se calculan añadiendo repetidamente la pendiente de la línea.

Puede que a usted le tiente probar esta técnica en BASIC mediante el empleo de la rutina Plotsub de la página 817. Pero este método tiene algunas dificultades:

- Si el valor de X2 es menor que el de X1, deberemos decrementar en lugar de incrementar.
- Si la pendiente es mayor que la unidad, la línea no será continua. Esto se debe a que Y quedará incrementado a su vez en más de uno a cada incremento de X.
- Las pendientes negativas son difíciles de implementar en código máquina. Se puede, sin duda, emplear el complemento a dos para representar los cambios negativos en X e Y, pero como la ordenada Y sólo alcanza valores de 0 a 199, y la X de 0 a 319, lo que necesitaremos serán complementos a dos de *dos bytes*.
- Las líneas verticales no pueden trazarse, pues el cálculo de sus pendientes implica una división por cero (X2 = X1 en las líneas verticales).

Algunos retoques

De los problemas enumerados, el primero se evita intercambiando los dos puntos extremos antes de los cálculos (el de partida se convierte en el de llegada, y viceversa).

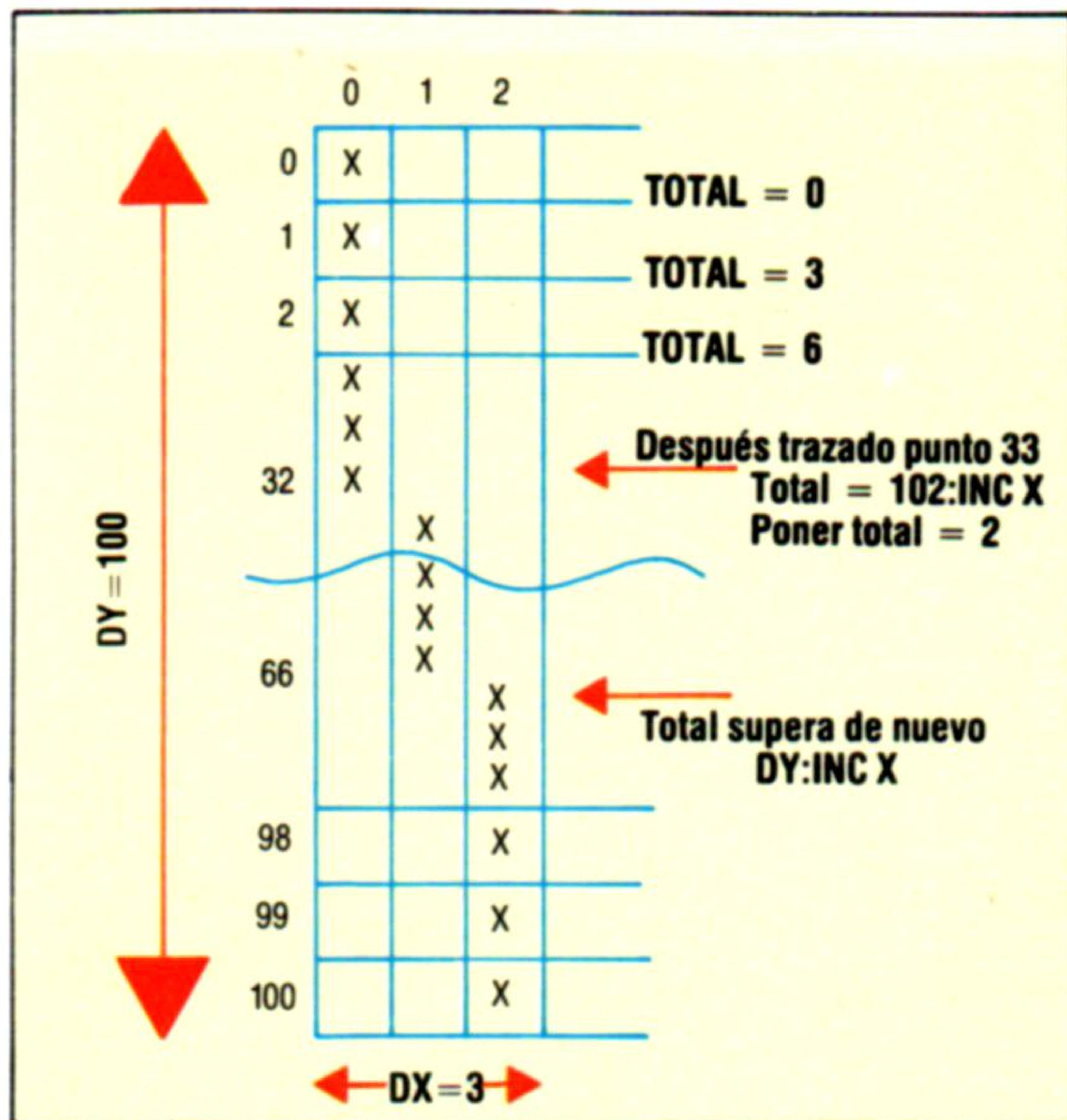
Para trazar una línea continua es necesario incrementar X y calcular Y si la pendiente es menor que uno, o bien incrementar Y y calcular X si la pendiente es mayor que uno. Con una bifurcación que sea consecuencia de tal condición (¿es P > 1?) evitamos usar la división y el complemento a dos de dos bytes.

Ya hemos ingeniado el procedimiento para trazar líneas con pendientes menores que uno. Consideremos ahora el caso de las pendientes que superan la unidad. Sean DX = X2 − X1 y DY = Y2 − Y1. Con sólo comparar DX y DY sabremos si la pendiente supera la unidad. Esto ocurrirá cuando DK < DY.

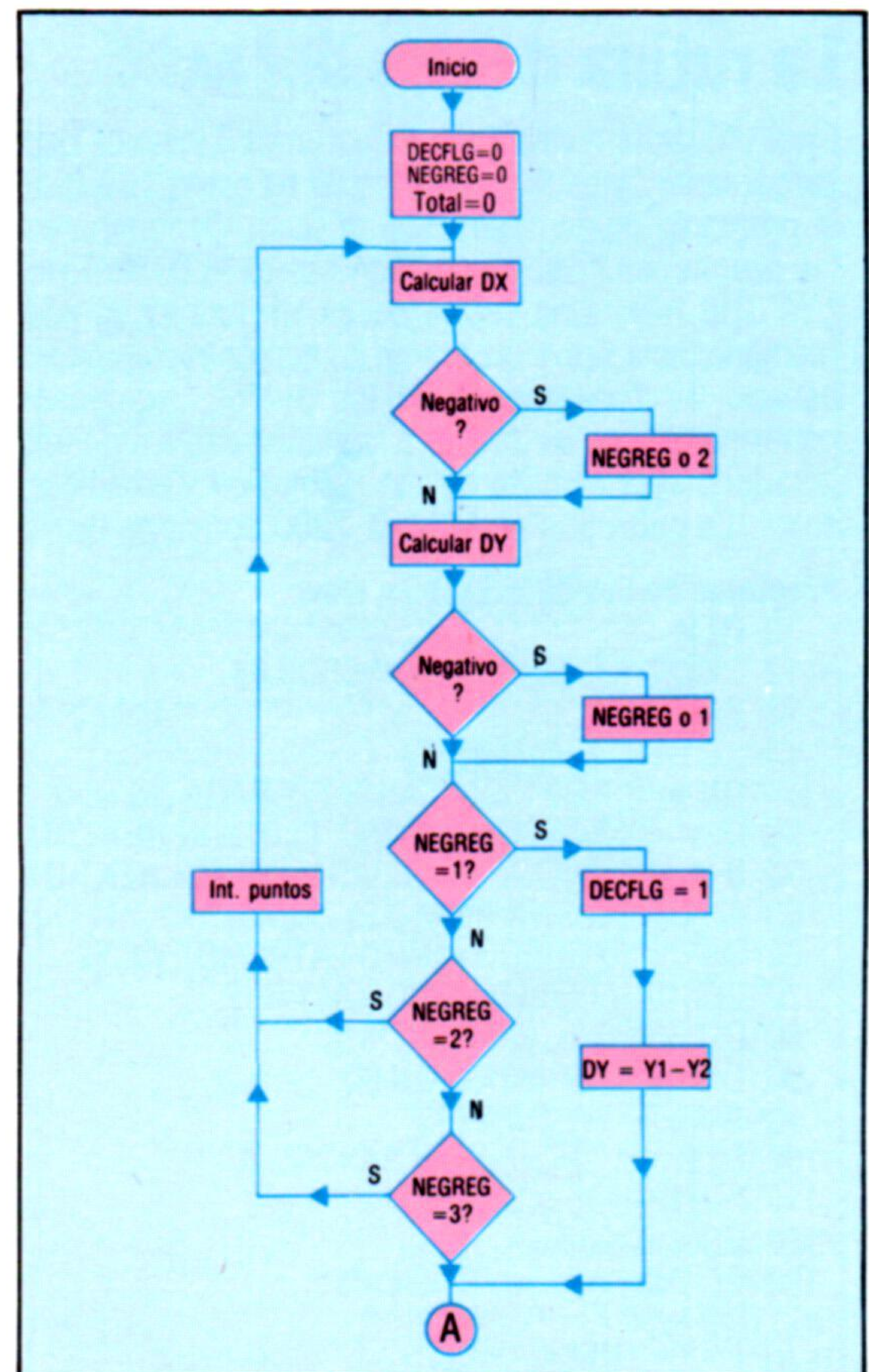
Para eludir ahora los valores negativos de DX y DY, el procedimiento es algo más complicado. Deberemos tener en cuenta cuatro casos posibles, siempre con la pendiente mayor que la unidad. La ilustración aparte describe estos casos. En nuestro programa debemos determinar con exactitud en qué caso nos encontramos para tomar, después, el camino más adecuado (suponiendo que deseamos mantener DX siempre positiva).

Caso a) Tomar DY como resultado de Y1 − Y2. Comenzando en (X1,Y1) iremos decrementando Y hasta que tome el valor Y2.
Caso b) Intercambiar los puntos y volver a empezar.
Caso c) Comenzaremos en (X1,Y1) incrementando Y hasta llegar a Y2.
Caso d) Intercambiar los puntos y volver a empezar.

Al comienzo de nuestro programa deberemos calcular DX y DY. Podemos colocar los bits en una determinada posición si DX o DY tienen valor ne-



Trama lineal
En el dibujo de la izquierda mostramos cómo puede trazarse una línea vertical u horizontal con exactitud en un campo de puntos discretos. No obstante, una línea en cualquier otro ángulo debe trazarse como una serie de escalones. La línea del dibujo une los puntos (0,0) y (2,100) mediante tres escalones verticales idénticos



Control lineal
El diagrama de flujo de la derecha muestra el método de la subdivisión de la línea previendo los cuatro casos en que la pendiente es mayor que uno. Las variables TOTAL y DECFLG (*flag* de *dec*remento) serán empleados por la rutina en el trazado que contempla el diagrama de flujo inferior. El trazado de puntos comienza en (X1,Y1) y dado que DX se mantiene positivo, la Y se incrementa o decrementa según el estado de DECFLG

gativo. El bit 0 de este registro (que llamaremos NEGREG) señalará el hecho de ser DY negativo, y el bit 1 indicará que DX es negativo. Seguidamente será consultado NEGREG para determinar en qué caso nos encontramos:

Caso	a	b	c	d
Valor de NEGREG	1	2	0	3
Equivalente binario	01	10	00	11

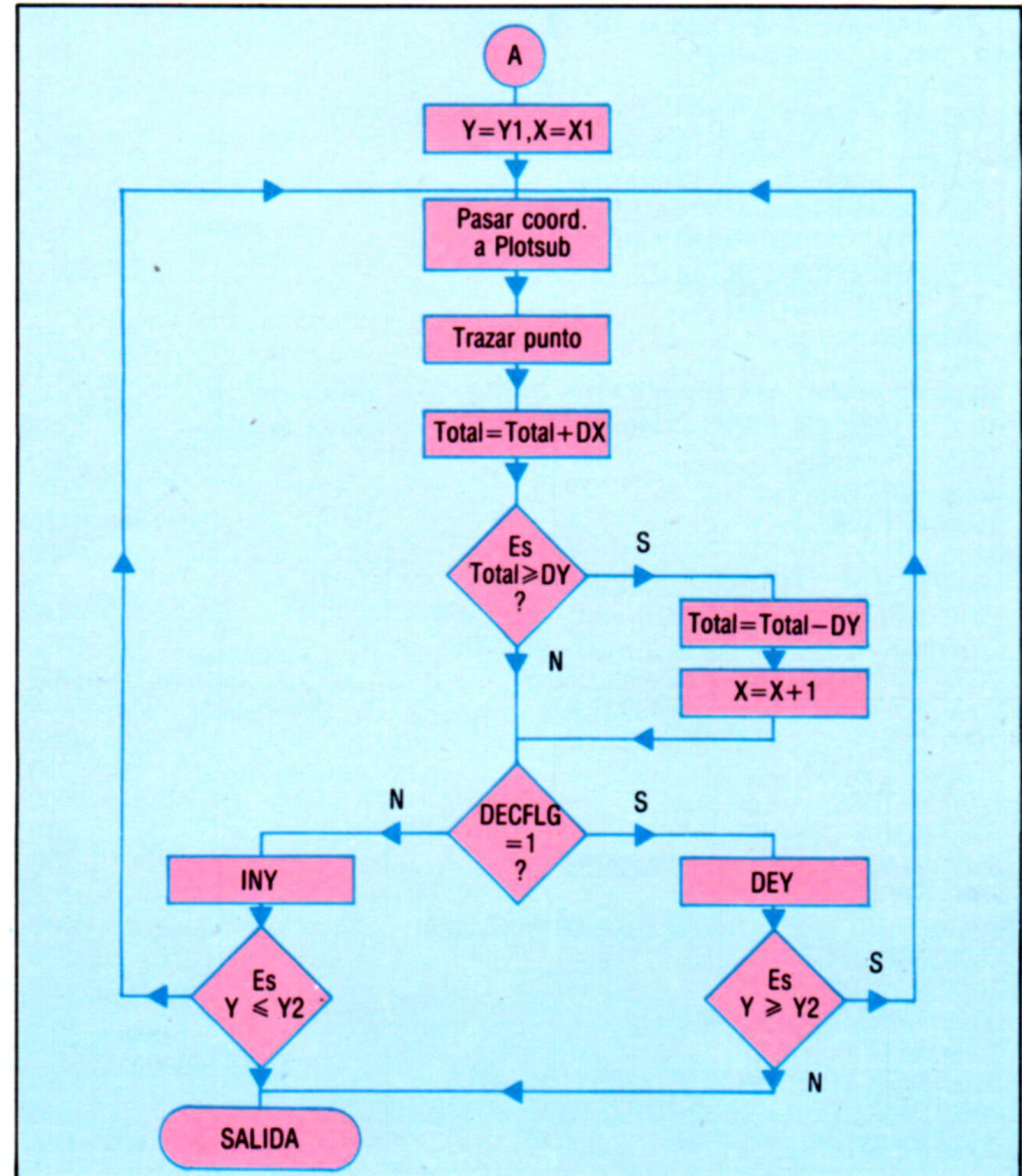
En cualquier caso debemos trazar la línea mediante el empleo de un bucle, comenzando en (X1,Y1), incrementando o decrementando Y según convenga y calculando el valor correspondiente de X antes de servirnos de la rutina Plotsub para trazar el punto. El bucle ha de terminar cuando Y haya alcanzado Y2.

De Y obtendremos X mediante una expresión reelaborada respecto de la primitiva ya explicada:

$$P = (Y-Y1)/(X-X1)$$
$$P(X-X1) = (Y-Y1)$$
$$(X-X1) = (Y-Y1)/P$$
$$X = X1 + (Y-Y1)/P$$

Cuando $(Y-Y1) = 1$, la expresión se simplifica: $X = X1 + 1/P$. Y puesto que $P = DY/DX$ tendremos que $X = X1 + DX/DY$. Pero si P es mayor que uno entonces DY será mayor que DX, de donde se deduce que la división DX/DY da siempre cociente 0 y resto DX. Tomando un total actualizado de los restos a medida que se itera el bucle mencionado, sólo debemos incrementar X cada vez que la suma de los restos sea mayor que DY. El total actualizado se vuelve a reinicializar a continuación.

Esto puede parecer algo complicado, pero veamos el método en acción considerando un ejemplo donde (X1,Y1) = (0,0) y (X2,Y2) = (2,100). La línea se visualizará como compuesta por una serie de tres barras verticales. Una línea bastante escarpada por cierto, pero si disminuimos la longitud de las barras la línea será más compacta y cuando la pendiente sea uno cada barra equivaldrá a un pixel de longitud, dando la apariencia de una perfecta diagonal.



La rutina dentro del BASIC

Para utilizar la rutina (que llamaremos Linesub) deberá cargar tanto Linesub como Plotsub tal como se indica en el programa de demostración en BASIC. Denominaremos los programas finales en código objeto PLOTSUB.HEX y LINESUB.HEX. Una vez cargadas las rutinas en código máquina, este breve programa de demostración pedirá al usuario las coordenadas de los puntos, examinará la pendiente y si no es inferior a la unidad empleará Plotsub para limpiar la pantalla en alta resolución y establecer el color. La subrutina en la línea 2000 convierte las X en byte *lo* y byte *hi* antes de colocar (POKE) estos valores en posiciones separadas dentro de Linesub para los extremos de la línea. En la línea 3000 la subrutina imprime los valores de todas las posiciones utilizadas para las variables por Linesub.

Como una opción al mecanografiado del código fuente de Linesub y al ensamblaje de código máquina, el programa cargador de código máquina introduce este mismo programa en la memoria "leyendo" (READ) una serie de valores (DATA) y colocándolos (POKE) en la memoria. Escriba este programa y ejecútelo (RUN) para que Linesub quede cargado en la memoria.

Programa de demostración en BASIC

```
10 REM ************************************
12 REM **            LINESUB 64            **
13 REM ************************************
14 :
15 DN = 8:REM PARA CASSETTE HAGA DN = 1
20 IF A = 0 THENA = 1:LOAD"PLOTSUB.HEX",DN,1
30 IF A = 1 THENA = 2:LOAD"LINESUB.HEX",DN,1
50 INPUT"PRIMER PUNTO EXTREMO";X1,Y1
60 INPUT"SEGUNDO PUNTO EXTREMO";X2,Y2
70 GOSUB1000: REM MODO ALT RES
80 GOSUB2000: REM LINESUB
90 GETA$:IFA$ = ""THEN 90
95 IF A$ = " "THEN200
100 REM **** PREPARAR PANTALLA ****
110 POKE49408,0:SYS 49422
120 GOSUB3000
125 GETA$:IFA$ = ""THEN125
127 GOTO50
140 :
200 REM **** TRAZAR TRIANGULO ****
205 XA = 30:YA = 10:XB = 310:YB = 98
210 XC = 90:YC = 180
220 GOSUB1000
230 X1 = XA:Y1 = YA:X2 = XB:Y2 = YB:GOSUB2000
240 X1 = XC:Y1 = YC:GOSUB2000
250 X2 = XA:Y2 = YA:GOSUB2000
255 GETA$:IFA$ = "" THEN 255
260 REM **** PREPARAR PANTALLA ****
270 POKE49408,0:SYS 49422
275 PRINTCHR$(147)
280 END
290 :
1000 REM **** USAR ALT RES ****
1010 POKE49408,1:POKE49409,1
1015 POKE49410,1
1020 SYS 49422
1030 RETURN
1040 :
2000 REM **** ENTRADA LINESUB ****
2010 MHI = INT(X1/256):MLO = X1-256*MHI
2020 NHI = INT(X2/256):NLO = X2-256*NHI
2030 POKE49920,MLO:POKE49921,MHI
2040 POKE49922,NLO:POKE49923,NHI
2050 POKE49924,Y1:POKE49925,Y2
2060 SYS 49934
2070 RETURN
2080 :
3000 REM **** IMPRIMIR VALORES ****
3001 RESTORE
3002 PRINTCHR$(147):REM LIMPIAR PANTALLA
3005 FORI = 0TO13
3010 READA$
3020 PRINTA$,PEEK(49920+I)
3030 NEXT I
3040 DATA X1LO,XIHI,X2LO,X2HI,Y1,Y2,DXLO
3050 DATA DXHI,DY,TEMP,TOTLO,TOTHI,NEGREG,DECFLG
3060 RETURN
```

Cargador de código máquina

```
10 FOR I = 49934 TO 50371
20 READA:CC = CC+A
30 POKEI,A:NEXT
50 READA:IFCC<>A THEN PRINT"ERROR SUMA
   COMPROBACION":END
100 DATA72,138,72,152,72,169,0,141,13
110 DATA195,141,12,195,141,10,195,141
120 DATA11,195,173,2,195,56,237,0,195
130 DATA141,6,195,173,3,195,237,1,195
140 DATA141,7,195,16,8,173,12,195,9,2
150 DATA141,12,195,173,5,195,56,237,4
160 DATA195,141,8,195,176,8,173,12,195
170 DATA9,1,141,12,195,173,12,195,201
180 DATA1,240,20,201,2,208,6,32,140
190 DATA196,76,19,195,201,3,208,19,32
200 DATA140,196,76,19,195,173,4,195,56
210 DATA237,5,195,141,8,195,238,13,195
220 DATA173,6,195,24,105,1,141,6,195
230 DATA173,7,195,105,0,141,7,195,238
240 DATA8,195,173,4,195,168,173,7,195
250 DATA201,1,240,115,173,6,195,205,8
260 DATA195,176,107,173,0,195,141,3
270 DATA193,173,1,195,141,4,193,152
280 DATA141,5,193,32,131,193,173,10
290 DATA195,24,109,6,195,176,8,141,10
300 DATA195,205,8,195,144,24,56,237,8
310 DATA195,141,10,195,173,0,195,24
320 DATA105,1,141,0,195,173,1,195,105
330 DATA0,141,1,195,173,13,195,201,1
340 DATA208,31,136,204,5,195,240,3,76
350 DATA161,195,152,141,5,193,173,0
360 DATA195,141,3,193,173,1,195,141,4
370 DATA193,32,131,193,76,134,196,200
380 DATA204,5,195,144,152,76,134,196
390 DATA173,0,195,141,3,193,173,1,195
400 DATA141,4,193,152,141,5,193,32,131
410 DATA193,173,10,195,24,109,8,195
420 DATA141,10,195,173,11,195,105,0
430 DATA141,11,195,173,10,195,56,237,6
440 DATA195,141,10,195,173,11,195,237
450 DATA7,195,141,11,195,48,15,173,13
460 DATA195,201,1,240,4,200,76,104,196
470 DATA136,76,104,196,173,10,195,24
480 DATA109,6,195,141,10,195,173,11
490 DATA195,109,7,195,141,11,195,173,0
500 DATA195,24,105,1,141,0,195,173,1
510 DATA195,105,0,141,1,195,205,3,195
520 DATA208,142,173,0,195,205,2,195
530 DATA208,134,104,168,104,170,104,96
540 DATA173,2,195,141,9,195,173,0,195
550 DATA141,2,195,173,9,195,141,0,195
560 DATA173,3,195,141,9,195,173,1,195
570 DATA141,3,195,173,9,195,141,1,195
580 DATA173,5,195,141,9,195,173,4,195
590 DATA141,5,195,173,9,195,141,4,195
600 DATA96,230
610 DATA50794:REM*SUMA COMPROBACION*
```

Listado assembly

```
                ;****************************************
                ;****************************************
                ;**                                    **
                ;**            LINESUB 64              **
                ;**                                    **
                ;****************************************
                ;****************************************
                ;
                PLTSUB  = $C183
                XLO     = $C103
                XHI     = $C104
                YLO     = $C105
                *       = $C300
                ;
                ;**** VARIABLES DE LINESUB ****
                ;
                X1LO    * = * + 1
                X1HI    * = * + 1
                X2LO    * = * + 1
                X2HI    * = * + 1
                Y1      * = * + 1
                Y2      * = * + 1
                DXLO    * = * + 1
                DXHI    * = * + 1
                DY      * = * + 1
                TEMP    * = * + 1
                TOTLO   * = * + 1
                TOTHI   * = * + 1
                NEGREG  * = * + 1
                DECFLG  * = * + 1
                ;
                ;**** LLEVAR REGISTROS A LA PILA ****
                ;
48                      PHA
8A                      TXA
48                      PHA
98                      TYA
48                      PHA
                ;
                ;**** INICIALIZAR VARIABLES ****
                ;
                START
A9 00                   LDA  #$00
8D 0D C3                STA  DECFLG
8D 0C C3                STA  NEGREG
8D 0A                   STA  TOTLO
8D 0B C3                STA  TOTHI
                ;
                ;**** CALCULAR DX ****
AD 02 C3                LDA  X2LO
38                      SEC
ED 00 C3                SBC  K1LO
8D 06 C3                STA  DXLO
AD 03 C3                LDA  X2HI
4C 86 C4                JMP  FINISH
                ;
                NODEY
C8                      INP
CC 05 C3                CPY  Y2
90 98                   BCC  PLOT
                ;
4C 86 C4                JMP  FINISH
                ;
                ;**** PENDIENTES MENORES DE 1 ****
                ;
                GLESS
AD 00 C3                LDA  X1LO
8D 03 C1                STA  XLO
AD 01 C3                LDA  X1HI
8D 04 C1                STA  XHI
98                      TYA
8D 05 C1                STA  YLO
20 83 C1                JSR  PLTSUB
                ;
AD 0A C3                LDA  TOTLO
18                      CLC
6D 08 C3                ADC  DY          ; TOTAL = TOTAL+DY
8D 0A C3                STA  TOTLO
AD 0B C3                LDA  TOTHI
69 00                   ADC  #$00
8D 0B C3                STA  TOTHI       ; COGER ARRASTRE
                ;
AD 0A C3                LDA  TOTLO
38                      SEC
ED 06 C3                SBC  DXLO
8D 0A C3                STA  TOTLO       ; TOTAL = TOTAL-DX
AD 0B C3                LDA  TOTHI
ED 07 C3                SBC  DXHI
8D 0B C3                STA  TOTHI
30 0F                   BMI  ADD         ;RESULTADO NEGATIVO
                ;
                ;
AD 0D C3                LDA  DECFLG
C9 01                   CMP  #$01
F0 04                   BEQ  DECY
C8                      INY
4C 68 C4                JMP  PAST
                DECY
88                      DEY
4C 68 C4                JMP  PAST
                ;
                ADD
AD 0A C3                LDA  TOTLO
18                      CLC
6D 06 C3                ADC  DXLO        ;SUMAR DX
8D 0A C3                STA  TOTLO
AD 0B C3                LDA  TOTHI
6D 07 C3                RDC  DXHI
8D 0B C3                STA  TOTHI
ED 01 C3                SBC  X1HI
8D 07 C3                STA  DXHI
10 08                   BPL  NBRWDX
AD 0C C3                LDA  NEGREG
09 02                   ORA  #$02        ;FLAG DX NEGATIVO
8D 0C C3                STA  NEGREG
                ;
                ;**** CALCULAR DY ****
                ;
                NBRWDX
AD 05 C3                LDA  Y2
38                      SEC
ED 04 C3                SBC  Y1
8D 08 C3                STA  DY
B0 08                   BCS  NBRWDY
AD 0C C3                LDA  NEGREG
09 01                   ORA  #$01        ;FLAG DY NEGATIVO
8D 0C C3                STA  NEGREG
                NBRWDY
                ;
                ;**** COMPROBAR ESTADO DE NEGREG ****
                ;
AD 0C C3                LDA  NEGREG
C9 01                   CMP  #$01
F0 14                   BEQ  NEG1
                ;
C9 02                   CMP  #$02
D0 06                   BNE  FALSE
20 8C C4                JSR  SWOP
4C 13 C3                JMP  START
                ;
                FALSE
C9 03                   CMP  #$03
D0 13                   BNE  NEG0
20 8C C4                JSR  SWOP
4C 13 C3                JMP  START
                ;
                NEG1
AD 04 C3                LDA  Y1
38                      SEC
ED 05 C3                SBC  Y2          ;RECALCULAR DY
8D 08 C3                STA  DY          ;COMO Y1-Y2
EE 0D C3                INC  DECFLG      ;ESTABLECER DECFLG
                ;
                NEG0
AD 06 C3                LDA  DXLO
18                      CLC
69 01                   ADC  #$01        ; SUMAR 1 A DX
8D 06 C3                STA  DXLO
AD 07 C3                LDA  DXHI
69 00                   ADC  #$00        ; COGER ARRASTRE
8D 07 C3                STA  DXHI
EE 08 C3                INC  DY          ; SUMAR 1 A DY
AD 04 C3                LDA  Y1
A8                      TAY              ; HACER Y = Y1
                ;
                ;**** COMPROBAR SI DX > = DY ****
                ;
AD 07 C3                LDA  DXHI
C9 01                   CMP  #$01
F0 73                   BEQ  GLESS
AD 06 C3                LDA  DXLO
CD 08 C3                CMP  DY
B0 6B                   BCS  GLESS
                ;
                ;**** TRAZAR PUNTO ****
                ;
                PLOT
AD 00 C3                LDA  X1LO
8D 03 C1                STA  XLO         ;PASAR BYTE LO DEX
AD 01 C3                LDA  X1HI
8D 04 C1                STA  XHI         ;PASAR BYTE HI DEX
98                      TYA
8D 05 C1                STA  YLO         ;PASAR Y
20 83 C1                JSR  PLTSUB
AD 0A C3                LDA  TOTLO
18                      CLC
6D 06 C3                ADC  DXLO        ;TOTAL = TOTAL + DX
B0 08                   BCS  DOINX       ;EXCEDE LA SUMA 1 BYTE?
8D 0A C3                STA  TOTLO
CD 08 C3                CMP  DY
90 18                   BCC  NOINX       ;EL TOTAL ES DY?
                DOINX
38                      SEC
ED 08 C3                SBC  DY          ;TOTAL = TOTAL - DY
8D 0A C3                STA  TOTLO
AD 00 C3                LDA  X1LO
18                      CLC
69 01                   ADC  #$01
8D 00 C3                STA  X1LO        ;INCREMENTAR X1
AD 01 C3                LDA  X1HI
69 00                   ADC  #$00
8D 01 C3                STA  X1HI
                NOINX
AD 0D C3                LDA  DECFLG
C9 01                   CMP  #$01
D0 1F                   BNE  NODEY       ;DECFLG = 1?
88                      DEY
CC 05 C3                CPY  Y2
F0 03                   BEQ  FINAL       ;SI Y = Y2 TRAZAR ULTIMO
4C A1 C3                JMP  PLOT
                FINAL                    ;TRAZAR ULTIMO PUNTO
98                      TYA
8D 05 C1                STA  YLO
AD 00 C3                LDA  X1LO
8D 03 C1                STA  XLO
AD 01 C3                LDA  X1HI
8D 04 C1                STA  XHI
20 83 C1                JSR  PLTSUB
                ;
                PAST
AD 00 C3                LDA  X1LO
18                      CLC
69 01                   ADC  #$01
8D 00 C3                STA  X1LO        ; INCREMENTAR X
AD 01 C3                LDA  X1HI
69 00                   ADC  #$00
8D 01 C3                STA  X1HI        ; COGER ARRASTRE
                ;
CD 03 C3                CMP  X2HI
D0 8E                   BNE  GLESS
AD 00 C3                LDA  X1LO        ; HA ALCANZADO X a X2?
CD 02 C3                CMP  X2LO
D0 86                   BNE  GLESS
                ;
                ;**** SACAR REGISTROS DE LA PILA ****
                ;
                FINISH
68                      PLA
A8                      TAY
68                      PLA
AA                      TAX
68                      PLA
                ;
60                      RTS
                ;
                ;**** FIN DE PROGRAMA ****
                ;
                ;**** SUBRUTINA INTERCAMBIO PUNTOS ****
                ;
                SWOP
AD 02 C3                LDA  X2LO
8D 09 C3                STA  TEMP
AD 00 C3                LDA  X1LO
8D 02 C3                STA  X2LO
AD 09 C3                LDA  TEMP        ;INTERC. BYTES LO DE X
8D 00 C3                STA  X1LO
                ;
AD 03 C3                LDA  X2HI
8D 09 C3                STA  TEMP
AD 01 C3                LDA  X1HI
8D 03 C3                STA  X2HI
AD 09 C3                LDA  TEMP        ;INTERC. BYTES HI DEX
8D 01 C3                STA  X1HI
                ;
AD 05 C3                LDA  Y2
8D 09 C3                STA  TEMP
AD 04 C3                LDA  Y1
8D 05 C3                STA  Y2
AD 09 C3                LDA  TEMP        ;INTERCAMBIAR Y
8D 04 C3                STA  Y1
                ;
60                      RTS
                ;
                ****FIN SUBRUTINA ****
```

Rutina Linesub en código máquina

La rutina en código máquina traza líneas con pendientes igual a uno, o mayores-menores que uno. No hay problema alguno con las líneas verticales pues, con el método que hemos descrito, se evita la división por cero. Para líneas con pendiente menor que la unidad son válidos los principios explicados. Lo único que hace falta es incluir una comprobación sobre si DX es mayor que DY (o sea, si la pendiente es menor que uno) con su correspondiente bifurcación.

■ La rutina utiliza dos bytes para cada X, pues ésta puede superar el valor 255. Sin embargo, como Y no puede superar 199, cabe en un byte.

■ Las posiciones que Plotsub utiliza para guardar las coordenadas de los trazados son asignadas al comienzo del listado en assembly. También se especifica la dirección del inicio de Plotsub.

■ A los bits 0 y 1 de NEGREG se les asigna el valor apropiado según sea el resultado de DX y DY. Para acceder a cada bit sin afectar al resto se emplea la instrucción lógica ORA.

■ Dado que Y puede contenerse en un solo byte, el registro Y se emplea para retener su valor durante el trazado. Los valores del registro no son alterados por Plotsub.

Nótese bien: para asegurarse de que Plotsub puede utilizarse con Linesub, deberá realizarse el siguiente cambio en el listado que usted conoce del código fuente (véase p. 819). Se debe incluir una instrucción bifurcadora entre SBC REMX y TAX.

Un cambio equivalente se realizará en BASIC, en el programa de demostración en BASIC insertando estas líneas:

```
5010 FOR I = HRSFLG TO
HRSFLG + 314:READ A
5380 DATA 169,7,237,8,
193,240,6,170,14,13,193
5410 DATA 38944:
REM*SUMA COMPROBACION*
```

Una historia clásica

Artic se fundó hace tres años con un capital de 20 libras: hoy es una sólida empresa con planes de ampliación a nivel mundial

En 1980 un estudiante de 17 años llamado Richard Turner empezó a escribir software. Los primeros juegos que escribió fueron *Battleships* (Batallas navales) y *Star trek* (Viaje a las estrellas) para el ordenador ZX80. Optó por escribir juegos de estrategia en vez de juegos más populares de estilo recreativo debido a las limitaciones de la máquina. Como él mismo explica: "El ZX80 limpiaba la pantalla cada vez que algo se movía, de modo que los únicos juegos que uno podía crear eran juegos para pensar y no juegos recreativos, que sólo salieron realmente con el Spectrum".

Su primer gran éxito lo obtuvo con el juego *ZX chess*, que lanzó en el verano de 1981 en la primera "feria del micro ZX". Turner utilizó sus recursos al límite: "La noche anterior todavía estábamos copiando cassettes, usando siete ZX81 y colocándolas en bolsitas de plástico con instrucciones que habíamos reproducido con la fotocopiadora de la escuela". El *ZX chess* fue un gran éxito y, según Turner, ganó 1 500 libras en la feria.

A fines de ese mismo verano Artic Computing se convirtió en una sociedad anónima, pero quedó relegada a un segundo lugar cuando Turner aceptó una beca de la Ford Motor Company para estudiar ingeniería eléctrica en el Imperial College de Londres. Sus estudios duraron sólo un año, al final del cual decidió tomarse otro de descanso para dirigir la empresa. Pero jamás volvió a la Universidad.

En un principio, Artic se dirigía desde el dormitorio de Richard, en la casa de sus padres, en Hull; pero cuando la lista de software de la empresa llegó a incluir 93 títulos, Turner decidió que había llegado el momento de contar con unas oficinas propias. En junio de 1983 la empresa se trasladó a sus oficinas actuales de Brandesburton (Humberside). Se modernizó el catálogo y se contrató más personal. En la actualidad Artic tiene empleadas a 15 personas, incluyendo a tres personas de televentas y cinco programadores de dedicación exclusiva, quienes perciben un sueldo además de los royalties.

Artic tiene planeado abrir su propia cadena de tiendas minoristas distribuidas por toda Gran Bretaña. Las tiendas se llamarán Artic Software Stations y venderán no sólo los juegos de Artic sino también productos de otras empresas. La primera tienda se abrió en julio de 1984 en Acton (West London) y es también la sede social de la empresa en Londres. Hay que destacar que el establecimiento no está situado en un sector de gran movimiento comercial y que está bastante alejado del centro comercial del West End londinense. Al preguntársele el motivo por el cual había elegido ese lugar en particular, Jeff Raggett, director de marketing de Artic para Londres, replicó que "un local céntrico costaría 300 o 400 libras a la semana, y esta tienda nos sale por muchísimo menos, de modo que no nos vemos obligados a vender muchas cassettes para cubrir los gastos. Mucha gente nos ha criticado, nos ha dicho que estábamos locos por el hecho de abrir tiendas, pero al menos así sabemos lo que es vender y podemos hablar con la gente sobre cuáles son los juegos que les gustan".

Otra innovación en la comercialización son los exhibidores de Artic. Estas unidades son cajas exhibidoras con capacidad para 64 cassettes. Actualmente Artic se los está vendiendo a los agentes de prensa, posibilitando que el público compre su software a nivel local en vez de tener que acudir a los grandes detallistas. Jeff Raggett afirma que estos exhibidores están obteniendo un notable éxito.

Artic piensa dirigir la comercialización de sus productos en el exterior ella misma en la medida de lo posible, y en estos momentos la empresa está considerando la posibilidad de una ampliación a nivel europeo. Para el mercado de América del Norte Artic ha firmado un contrato con dos casas de software ya establecidas, Softsync y la International Publishing Corporation.

Hasta la fecha los mayores éxitos de ventas de Artic han sido *Bear Bovver* (del cual se han vendido más de 40 000 cassettes), *Galaxians* y *Gobbleman*. Recientemente la empresa ha lanzado un nuevo juego para el Spectrum llamado *World cup* (Copa del mundo), del que en sólo tres semanas se han vendido 5 000 ejemplares.



Ian McKinnell

Los juegos de Artic
Una muestra de los juegos más destacados de Artic, incluyendo los exitosos *Bear Bovver* y *World cup* (Copa del mundo), una versión para el Spectrum del popular deporte del fútbol

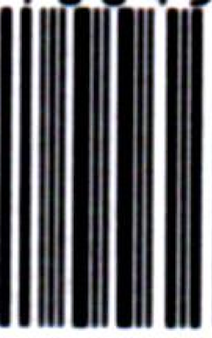


Richard Turner


Editorial Delta, S.A.
mi COMPUTER



9 788485 822836


10045